

# THE GAMES FACTORY 2 MULTIMEDIA FUSION 2

## USER'S MANUAL



### Health warning :

Some people experience epileptic seizures when viewing flashing lights or patterns. These persons may experience seizures while watching TV or playing video games. Consult your doctor before playing video games if you have an epileptic condition.

# License

You should read the following terms and conditions carefully before using this software. Your use of this software indicates your full acceptance of this license agreement and warranty.

1. **GRANTING OF LICENSE.** Clickteam, as Licensor, grants to you, the Licensee, a non-exclusive license to use this software program (hereinafter referred to as the «SOFTWARE») in accordance with the terms contained in this license.

You may use the SOFTWARE on a single computer (you may install the program on several machines provided that it is used simultaneously on a single machine). You may access the SOFTWARE through a network, provided that you have obtained individual licenses for the software to cover all workstations that will access the software through the network.

2. **NO REVERSE ENGINEERING.** You may not modify, disassemble or decompile the SOFTWARE or the stand-alone applications created with the SOFTWARE, in whole or part.

3. **STAND-ALONE APPLICATIONS.** The SOFTWARE may be used to create an unlimited number of stand-alone applications and screen savers. You are allowed to sell the stand-alone applications and screen savers you create with the SOFTWARE, provided that you agree with the Runtime Distribution Agreement. The Runtime Distribution Agreement is available in the online help on the CD-ROM and on the Clickteam Web site, <http://www.clickteam.com>.

4. **COPYRIGHT.** All title and copyrights in and to the SOFTWARE, including but not limited to any images, texts, and sounds incorporated into the SOFTWARE, are owned by Clickteam or its suppliers.

5. **REDISTRIBUTION.** You are NOT allowed to redistribute the SOFTWARE.

6. **GOVERNING LAW.** If you acquired this product in France, this agreement is governed by the French laws. If you acquired this product outside France, then local law may apply.

7. **DISCLAIMER OF WARRANTY.** THE SOFTWARE AND THE ACCOMPANYING FILES ARE PROVIDED «AS IS» WITHOUT WARRANTY OF ANY KIND. TO THE MAXIMUM EXTENT PERMITTED BY APPLICABLE LAW, CLICKTEAM DISCLAIMS ALL WARRANTIES, EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, ANY IMPLIED WARRANTIES OF PERFORMANCE, MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, AND NONINFRINGEMENT. TO THE MAXIMUM EXTENT PERMITTED BY APPLICABLE LAW, IN NO EVENT SHALL CLICKTEAM BE LIABLE FOR ANY DIRECT, INDIRECT, CONSEQUENTIAL OR INCIDENTAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, DAMAGES FOR LOSS OF BUSINESS PROFITS, BUSINESS INTERRUPTION OR LOSS OF BUSINESS INFORMATION) ARISING OUT OF THE USE OF OR INABILITY TO USE THE SOFTWARE, EVEN IF CLICKTEAM HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

# Summary

|  |           |
|--|-----------|
| <b>License</b>                                 | <b>3</b>  |
| <b>SUMMARY</b>                                 | <b>4</b>  |
| <b>Credits</b>                                 | <b>5</b>  |
| <b>Welcome</b>                                 | <b>6</b>  |
| <br>   |           |
| <b>The Main Interface</b>                      | <b>7</b>  |
| 1. Edit time Vs Run time                       | 8         |
| 2. The Workspace Toolbar                       | 8         |
| 3. The Property Toolbar                        | 12        |
| 4. The Storyboard Editor                       | 13        |
| 5. The Frame Editor                            | 15        |
| 6. The Objects                                 | 21        |
| 6.1. The Picture Editor                        | 21        |
| 6.2. The Animation Editor                      | 24        |
| 6.3. The properties of the Active Object       | 29        |
| 6.4. Object inventory                          | 40        |
| 7. The Event Editor                            | 43        |
| 8. The Expression Editor                       | 58        |
| 9. The Event-list Editor                       | 62        |
| <br>   |           |
| <b>Appendices</b>                              | <b>65</b> |
| 1. Global events and behaviors                 | 65        |
| 2. The qualifiers                              | 67        |
| 3. Fast loops                                  | 68        |
| 4. Creating a scrolling game                   | 69        |
| 5. Editing the data elements of an application | 71        |
| 6. Extensions                                  | 73        |
| 7. The runtime options                         | 75        |
| 8. The debugger                                | 77        |
| 9. Creating your standalone application        | 79        |
| <br>   |           |
| <b>Notes</b>                                   | <b>81</b> |



© 1996-2006, Clickteam. All rights reserved.

**This program was created and programmed by**

Yves Lamoureux & François Lionet

**Graphics by**

Nicolas Sourdeval

**Clickteam is**

D.T. Holder

Francis Poulain

Francois Lionet

Jason Darby

Jeff Vance

Rhon Schlick

Yves Lamoureux

**Special thanks to**

Anders Riggelsen  
Andrew Kourosh  
Andrew Mather  
Andy Hewitt  
Ashley Gullen  
Chris Branch  
Chris Newhouse  
Chris Street  
David Clark  
David Evans  
Daymian Tiedemann  
Donald May  
Erik Gustafsson

Eva Isotalo  
Gareth Ben Martin  
Gareth Lowe  
Gayle Poirier  
Geoff & June  
Jamie McLaughlin  
Joseph Wojciechowski  
Josh Downen  
Josh Whelchel  
Lior Halphon  
Marcus Henrikson  
Mattis Lorentzon  
Melli Georgiou

Olivier Behr  
Paul Boland  
Peter Heinze  
Philippe Cizaire  
Rando Wiltschek  
Raymond Emonnot  
Rich Whilding  
Robbie Shields  
Ryan Sadwick  
Sarah Greenland  
Scott Watson  
Sean Poling

## Welcome to **Multimedia Fusion 2** and **The Games Factory 2!**

This manual has been written for both Multimedia Fusion 2 and The Games Factory 2: these products share a number of common features. Within the manual we will be identifying both products as Multimedia Fusion (or its abbreviation of MMF). Any features specified will apply to both products unless otherwise indicated.

This manual also applies to the MMF2 Developer version. MMF2 Developer adds many features and capabilities while providing a smooth transition from MMF2 Standard and The Games Factory

This manual will review in details the most important parts of the product:

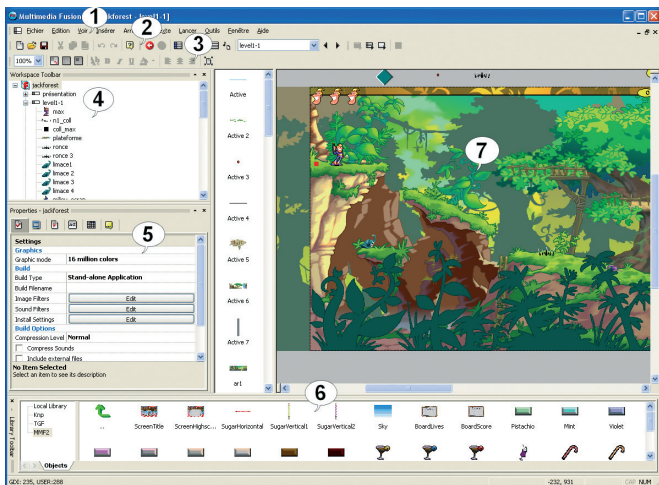
- The **interface**: the workspace toolbar, the property explorer
- The **editors**: the storyboard editor, the frame editor, the picture and animation editors
- The **objects**: active, string, counter etc. objects
- The **event editor**, and the event programming
- And finally in appendixes various considerations about Multimedia Fusion.

## The Chocobreak tutorial

Before reading this manual, if you are new to Multimedia Fusion, we strongly suggest that you go through the Chocobreak tutorial. This tutorial will take around one hour of your time to complete, and will give you a basic working knowledge of Multimedia Fusion.

You will find other tutorials, for multimedia applications and games on the CDROM in the tutorial folder.

# The main interface



The picture above represents The Multimedia Fusion interface. The main interface parts are as follows:

1. **The menu bar.** This contains all the options available for you through out MMF.
2. **The toolbars.** These contain icons for quick access to MMF (Multimedia Fusion) functions.
3. **The editor toolbar.** This toolbar changes depending on the current editor, and allows a quick access to the editor functions.
4. **The project window.** This very important part of the interface presents the current project or application in detail, in a structured way.
5. **The property explorer.** This window shows all the options available for the currently selected object.
6. **The library window.** This show the content of the libraries. (A library contain graphics and/or objects ready for you to use).
7. **The main editor screen.** This area displays the currently selected editor.

# 1. Edit time Vs Run time

Multimedia Fusion is an application used to **create applications** and used in the following two modes:

- **Edit time.** This is when you create your application, drop the objects, and enter the events. This is the design time.
- **Run time.** (referred to as Runtime) Once your application is ready to work, or during testing, you can launch the runtime and the objects come to life as a full program. The runtime can be launched via the buttons of the Run toolbar:

From left to right:

- **Run project.** Launches the first application of your project.
- **Run application.** Launches the current application.
- **Run frame.** Available if you are editing a frame, run the frame only.



You will find the same options in the “Run” submenu of the main menu.

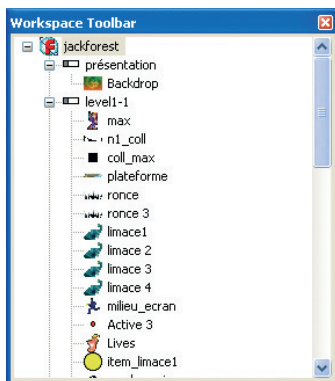
## 2. The Workspace Toolbar

We will start our description of the interface by concentrating on the workspace toolbar. The workspace toolbar is the area where you can see the entire structure of your application. It displays:

- The **application**
- The **frames**
- The **objects**

### What are these items?

- The **application** is the program you are creating and represents the file on the hard disc. Note: Multimedia Fusion applications are saved with the suffix “.MFA”. You edit the content of the application with the **storyboard editor**. In the previous picture, the application name is “Zeb”.
- The **frames** are the different screens of your application. If your application is a game, then a frame might correspond to a level of the game. A game with three frames could contain as first frame, the title page, as



second page the game itself, and as third page a high-score table. You edit the content of the frame in the frame editor. In the previous picture, the frames are named “Version screen”, “Zeb title screen” and “Funky Jungle 1-1”.

- The **objects**. The objects are the items contained in a frame. For a game, objects can be a ball, the main player character, a trap etc. For multimedia presentations, an object can be a string of text, a text edit zone, a button, a list-box etc. Of course any type of application can use any type of object. All the objects can be edited in the frame editor. In the above picture, the objects are named “Zeb”, “Quick backdrop 2”, “Real-Zeb” etc.

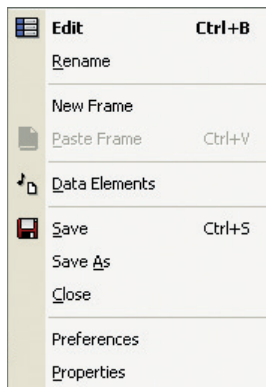
As you can see the structure of the application is displayed using a parent / child relationship in the workspace toolbar. I.e: Application / Frame / Objects. An application may have any multiple frames with each frame having multiple object.

## Workspace toolbar menu options

By clicking with the right mouse button on the items contained in the workspace toolbar, you can access different parts of MMF and perform some editing functions.

Right clicking on the **application** icon opens the following menu:

- **Edit**. Opens the storyboard editor to edit the layout of the frames of the application.
- **Rename**. Renames the application.
- **New frame**. Inserts a new, blank frame in the list of frames.
- **Paste frame**. If a frame is contained in the clipboard, pastes the frame in the list of frames.
- **Data elements**. Opens the Data Elements editor. For more information on this powerful option, please refer to appendix 5.
- **Save**. Saves the application on disc using the current name.
- **Save As**. Saves the application on disc with a new name.
- **Close**. Closes the application.
- **Preferences**. Opens the preferences dialog box.
- **Properties**. Displays the properties of the application in the property toolbar.









Right clicking on the **frame** icon opens the following menu:

- **Frame editor.** Opens the frame editor.




This option is in bold, it indicates that the Frame editor option is the default option: it is activated when you double click on the frame. You can change this option in the preference dialog.

- **Event editor.** Opens the event editor. Please refer to chapter 8 for more information on the event editor.
- **Event list editor.** Opens the event list editor. Please refer to chapter 9 for more information.
- **New folder.** Creates a new object folder. This option is the same as the one you find in the frame editor: it creates a new object folder in which you can drag & drop the objects of your frame, to simplify the display.
- **New object.** Opens the new object dialog, the same dialog than the one found in the frame editor
- **Rename.** Allows you to rename the selected frame.
- **Cut / Copy / Paste / Delete :** the classic edit options, that work on frames.
- **Preferences.** Opens the preference dialog box.
- **Properties.** Displays the properties of the frame in the property toolbar.

|   |                     |               |
|---|---------------------|---------------|
|    | <b>Frame Editor</b> | <b>Ctrl+M</b> |
|  | Event Editor        | Ctrl+E        |
|  | Event List Editor   | Ctrl+L        |
| <hr/>   |                     |               |
|   | New Folder          |               |
|   | New Object          |               |
| <hr/>   |                     |               |
|   | Rename              |               |
| <hr/>   |                     |               |
|  | Cut                 | Ctrl+X        |
|  | Copy                | Ctrl+C        |
|  | Paste               | Ctrl+V        |
|   | Delete              | Suppr         |
| <hr/>   |                     |               |
|   | Preferences         |               |
|   | Properties          |               |

Right clicking on an **object** icon opens the following popup menu:

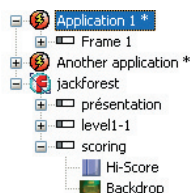
- **Edit.** Opens the editor for this object. The editor may vary from object to object, for example for Active objects, it will open the Animation editor. For the System box object, it will open the edit dialog box of this object. Many objects do not have an edit box, when all the necessary data is available through the property toolbar.
- **Rename.** Renames the object.
- **Cut / Copy / Paste / Delete.** Edit functions applicable to this object.
- **Clone.** Will create a new object identical to the object selected. The object will have a name different from the name of the original object.
- **Duplicate.** Creates a copy of the first object. The new object will be another instance of the same object.
- **Preferences.** Opens the preference dialog box.
- **Properties.** Displays the properties of the object in the property toolbar.

|   |             |        |
|---|-------------|--------|
|   | <b>Edit</b> |        |
| <hr/>   |             |        |
|   | Rename      |        |
| <hr/>   |             |        |
|   | Cut         | Ctrl+X |
|  | Copy        | Ctrl+C |
|  | Paste       | Ctrl+V |
|   | Delete      | Suppr  |
| <hr/>   |             |        |
|   | Clone       |        |
|   | Duplicate   |        |
| <hr/>   |             |        |
|   | Preferences |        |
|   | Properties  |        |

## Drag & drop, cutting and pasting

Drag & drop is possible in the workspace toolbar: simply click on the item you want to move and drop it to the new position. You can drag & drop whole applications, frames and objects.

The common edit function of cutting and pasting are possible as well. You can access them from the popup menu you get when clicking with the right mouse button on an item, or from the main menu. Frames and objects can be copied into the clipboard and then pasted wherever you want. You can cut objects from one frame and paste them in another frame. You can copy a complete frame from one application and copy it in another application.



## Applications and projects

It is possible to open more than one application at the same time in the workspace toolbar. Simply choose “Load file”, or “Create new” from the menu option as many time as you want. The applications are displayed one after the other in the workspace toolbar.

The picture shows a workspace toolbar with 3 applications loaded at the same time.

## The notion of projects

A project is a set of applications. For example, the first application could be the title of your multimedia presentation, the second the main pages, and the third one the conclusion. MMF provides you with a way of working easily with more than one application and allows the developer to keep related applications together.

To create a project, simply load your applications in the workspace toolbar, and choose the menu option “**File / Save project**” enter a name and click on OK. A new file will be created, with the “MFW” extension. Next time you use MMF, do not load individually each application, but load the MFW file: all the applications will be loaded automatically.

The first application displayed in the workspace toolbar is called the Main application: it is the first application to be run when you click on Run Project.

Projects are only present in Multimedia Fusion.

## Re-opening the workspace toolbar

If by mistake you close the workspace toolbar, you can re-open it by choosing the main menu option: “**View / Toolbars / Workspace**”

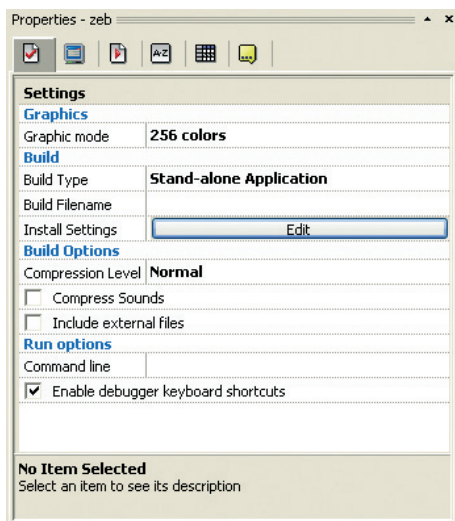
### 3. The Property Toolbar

The property toolbar is an essential part of Multimedia Fusion: it provides the interface to choose all the necessary options for the applications, frames and objects.

Using the property toolbar is easy because everything is automatic. The property toolbar adapts to display the properties of the selected object. Select the application in the workspace toolbar, and the application properties will be displayed. Select a frame in the storyboard editor, and the frame properties will be displayed. Select an object in the frame editor, and the object properties will be displayed.

The picture below presents a page of the properties of an application. It is divided in three zones:

- The top of the display presents a set of **tabs** each representing a different chapter of properties. For example, for applications you have the “**Settings**” tab, which contains different settings like the number of colors, the installer settings etc; the “**Window**” tab which contains everything dealing with the application window, the “**Runtime options**” to adjust the run-time environment of your application; the “**Values**” tab to set-up internal variables; the “**Events**” tab for global events, and the “**About**” tab for disclaimer type information.




- The medium section of the window contains the **properties** themselves. Each property is displayed on one line, and should be self-explanatory. Check boxes, edit zones, color selectors, combo boxes: you may find all of them and more in the properties. Try and experiment with them, it can do no harm to try.

- The bottom of the window is the **information panel**: it shows the name of the selected item and a description of its function.

## 4. The Storyboard Editor
















The **storyboard editor** is the place where you edit and create the different frames (screens) of the application. Although you can do that in the workspace toolbar, it may be simpler for some people to use the storyboard editor.

### How to open the storyboard editor?

- Double click on the application name in the workspace toolbar (or choose the Edit option in the popup menu)
- Click on the Storyboard editor icon  in the Navigation bar
- Press the keys **Control-B**



### The display of the storyboard



The storyboard editor shows specific details for every frame of your application. The display is organised in columns and lines. The columns are the following :

| No. | Thumbnail   | Comments   |
|-----|---|--|
| 1   |  | <div>Title : présentation</div> <div>Password :</div> <div>    640 by 480</div>            |
| 2   |  | <div>Title : level1-1</div> <div>Password :</div> <div>    1440 by 480</div>               |
| 3   |  | <div>Door</div> <div>Title : scoring</div> <div>Password :</div> <div>    640 by 480</div> |

- **Number:** shows the number of the frame in the application
- **Thumbnail:** shows a reduced version of the frame
- **Comments:** displays information about the frame.

Each line corresponds to the specifics of a given frame. The comments section contains several important data:

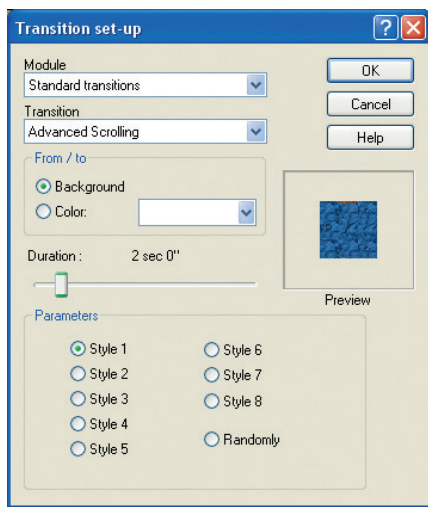
- **Title:** the name of the frame. Click on the name to rename it.
- **Password:** you can define a password for each frame. A menu option will be available during runtime allowing the user to enter a password. If the password matches the one you have defined, the application will run the corresponding frame.
-  indicates a Multimedia Frame
-  fade-in transition icon. This icon will be activated if a fade-in transition is defined. Click on the icon to insert a transition.

-  fade-out transition. Same as above, but the transition is used when the frame quits. Click on the icon to insert a transition.
-  Shows the size of the frame, in pixels. Just click on the number to resize the frame. This size is the same as the one displayed in the Size property of the frame.

## Inserting a fade transition

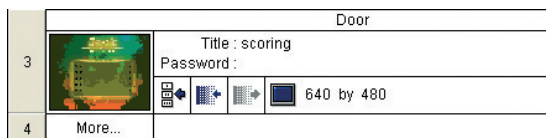
Fade in and out transitions are used to make various effects between frames. The fade-in transition makes the transition between the previous frame and the current one (the display of the previous frame is slowly replaced by the current frame), and the fade-out transitions slowly replaces the content of the current frame by a plain color. Transitions are important for multimedia presentation where you need to smoothly go from one panel to another.

To insert a new fade, click on the fade button in the “Contents” column of the storyboard editor. This will open the transition set-up dialog box:



The different zones of this dialog box are the following:

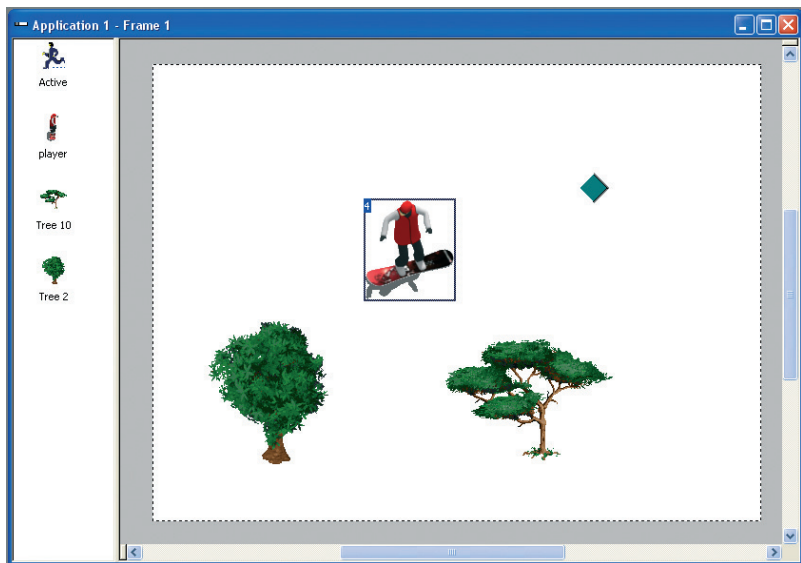
- **Module.** Transitions come as modules, one module containing several transitions. Choose the module with this combo-box.
- **Transition.** This combo-box shows the available transitions in the selected module.
- **From / to.** For fade-in transitions, indicates the origin screen: it can be the Background of the previous frame, or a plain Color, in this case you can select it in the color selector. For fade-out transition, indicates a color.
- **Duration.** Sets the speed of the transition.
- **Preview.** Shows the transition in action.
- **Parameters.** This section of the dialog box varies accordingly to the transition chosen. It usually displays the different styles available for the transition.



Once a transition is defined, it is displayed in the storyboard editor as a new line before or after the frame:

## 5. The Frame Editor

The frame editor is where you edit the content of the **frames** of your application. It allows you to drop the different objects in the frame, and set their respective positions, size, etc. The frame editor edits one frame at a time.



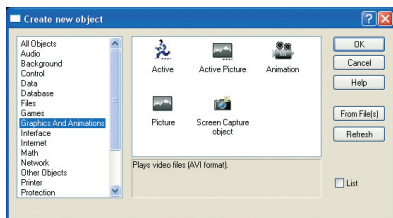
The left panel of the window shows as list of the objects contained in the frame. By right clicking on the left panel you can choose to display the objects as small or big icons, and to sort them by size or by name.

The main panel shows the frame itself. The grey areas represent the area outside the frame and can be thought of as active memory. It is possible to position object in the grey are but they won't appear in runtime as they are outside of the view area. You can use the outside as a docking space to store the objects you may need during your application, object that will move into the view area at runtime, or for object that may be needed but not seen at runtime.

## Creating a new object

You can create a new object from scratch. To do so, you have two options:

- **Right click** on the frame and choose the **Insert object** option
- Open the main menu and choose the option **“Insert / New object”**
- Double click on an empty zone in the frame



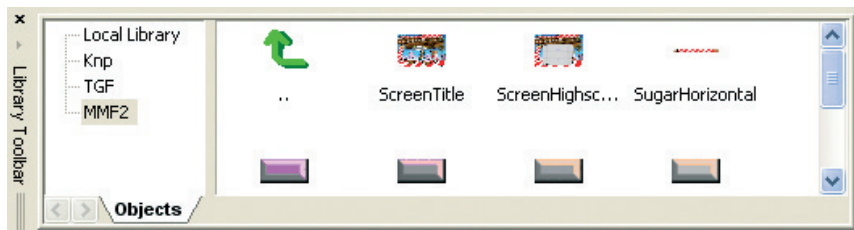
It will open the **“New object”** dialog box.

The dialog box is organised in two panels. The left panel shows the different categories of objects. As you can see a lot of categories are available. Click on the chosen category to see the objects displayed in the right pane.

The **“From file”** button opens a file selector for you to choose an object in another application.

The **Refresh** button updates the list of objects from the disc: you might have copied a new extension object in your extension folder, click on Refresh to display it in the list.

To create the object, just double click on it, or select the object and click on the OK button. Once selected the dialog box will vanish and you only need to click somewhere on the frame to drop the object.



## Getting an object from a library

Multimedia Fusion comes with a large set of pre-made objects which are available in different libraries. To display the content of the libraries, you need to open the Library window if it is not already done: choose the option **“View / Toolbars / Library window”** in the main application menu, and a new window will popup in the bottom of your screen:

The library window is divided in two zones: the left panel shows the current directory containing the libraries. In our example, we have four different directories. Double click on a directory to open it. MMF displays in the right panel the name of all the available

libraries in the folder. Now you just have to click on one of the library to open it: the content of the right panel is replaced by the content of the library with all the objects.

A click on the “..” (previous) icon will re-display the list of libraries.

A double click on one object will display the object in real size in a small window. Just click anywhere to erase it.

To get the library object in your application, click on it, and drop it in the frame.

## Creating your own libraries and displaying them

You can of course create your own libraries: just create the objects you want in the first frame of a new application, and save the application in the “**Lib\Objects**” folder in the Multimedia Fusion folder in C:\Program Files. The new library will appear in the “**Local library**” section.

You can also create a new folder with your new libraries and display it in the window: right click somewhere in the left panel, and choose the “**New**” option. In the directory selector, choose the folder you want to display.

## Selecting the objects

To select an object in the frame editor, just click on it with the left mouse button. A selection border will appear around the object. A number is also displayed in the top-left corner: this number is the number of the object in the list of objects of the frame editor. This can be important when you are modifying the priorities of the objects.



If you click a second time on the same object, the border is replaced by the resizing border. Just grab one of the handles to resize the object.



If you click a third time on the object, the border is replaced by the rotation border.



Grab the handle and you can rotate the object. Warning, for complex objects, the rotation process can be quite long.

As soon as you click on an object, its properties are displayed in the property toolbar.

## Multiple selections

You can select more than one object at the same time: either hold the SHIFT key down while clicking on the objects, or draw a rectangle around the objects.

You will see that the colors of the selection rectangle are different. The first object of the selection is called the main object, and its border is drawn in red. The other objects are drawn in blue. The main object is important for certain order and size options of the frame editor. You can, if you want, change the main object by clicking on it with the CTRL key down.

If more than one object is selected at the same time, the property toolbar displays only the properties that are common to the objects selected. A change of one of the property will affect all the selected objects at once.

## The frame editor pop-up menu

When you click with the right mouse button somewhere in the frame editor, or on an object, Multimedia Fusion displays a pop-up menu.

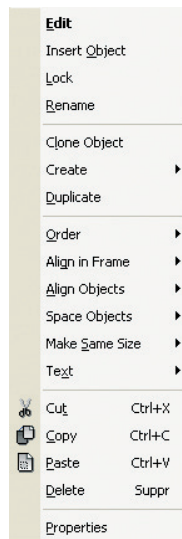
- **Edit.** Opens the edit dialog of the object. For example, for active object, it will open the animation editor.

- **Insert object.** Inserts a new object (opens the new object dialog).

- **Lock.** Locks the object: make it un-selectable. This option is very useful if you have a large number of objects one above the other. Locking objects that you do not want to select can save time and effort while using the Frame Editor. To unlock previously locked objects, you need to choose the “**Unlock all**” option from the main menu, in the “**Arrange**” sub-menu. To unlock a single object, you can press SHIFT + CTRL while clicking on it.

- **Rename.** Renames the object.

- **Clone object.** Creates new objects using the current object as a template. Warning, the created objects are different from the original objects. Use the duplicate option to create multiple copies of the same object. When you select this option, a small dialog is opened, allowing you to create more than one object in the form of an array.

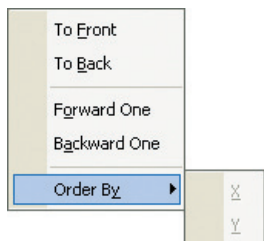


|                       |
|-----------------------|
| Quick Backdrop Object |
| Backdrop Object       |
| Active Object         |

- **Create submenu**

This option allows you to create different types of objects from the selected object. For example, you can create a backdrop object from an active object, an active object from a backdrop object, an active object from a string object.

• **Duplicate.** Creates multiple instances of the same object. The created objects are the same as the source object (they have the same name). Very useful for certain types of games, for example a Breakout game in which you need to create a wall of bricks.



#### • Order submenu

The order submenu allow you to set the display priority of the objects, ie. which object is in front of what object. **To front** and **To back** option are self-explanatory. **Forward one** and **Backward one** moves the object in the list of objects of the editor. The number indicated when you select an object is the indicator of the position of the object in the list. The lower the number, the more “in the back” the object will be.

**Order by X or Y** sorts the objects from their co-ordinates. For example, you can create a fake perspective effect if you sort the object on the Y co-ordinates: the objects closer to the bottom of the screen will be in front of the object positioned higher on the screen.

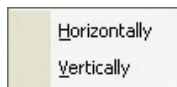


#### • Align in frame submenu

This submenu allows you to position the object at key places in the frame. For example, **Left** will put the objects on the left border of the frame.

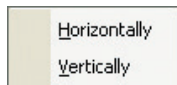
#### • Order objects submenu

This submenu is only activated if more than one object is selected at the same time. It allows you to align the objects together. The base position taken is the position of the main object selected (red border). For example, alignment on the top, will put all the objects at the same Y coordinate, the coordinate of the main object.



#### • Space object menu

These options are available if more than 3 objects are selected at the same time. They equalise the distance between the objects.

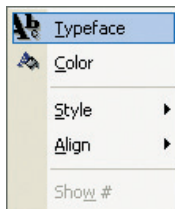


#### • Make same size submenu

These options are only valid if more than two objects are selected at the same time. It makes the size of the selected object equal to the size of the main object from the selection.

#### • Text submenu

This classic submenu is available for text objects, or objects that deal with text. It allows the modification of the casual options for text: color, font-change etc.



- **Cut / Copy / Paste / Delete.** The usual edit options are present in this menu. You can copy any kind of object to the clipboard and paste it anywhere as you want.
- **Properties.** Displays the properties of the selected object.

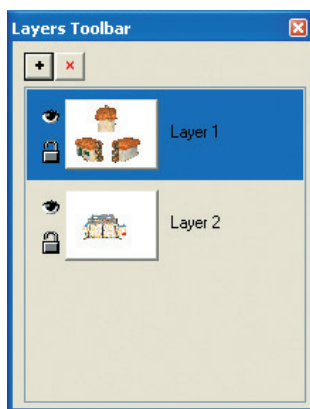
Most of the options of the pop-up menu are available in the main MMF menu, and some of them are available in the Frame editor's toolbar.

## Using the layers

The Frame editor allows the use of **layers**. Layers are useful to create multiple planes of objects, or perspective objects. Layers are also very useful for **parallax scrolling games**, please read the appendix 4 about scrolling games.

To show the Layer toolbar, select the main menu option “**View / Toolbars / Layer toolbar**”. The tool bar appears at the right of the frame editor's window.

The picture below shows a layer toolbar with two layers defined. When you drop an object in the frame editor, it is inserted in the selected layer. If the selected layer is above all other layers, then the object will be in front of the other objects. Within a layer, the order of the objects works as usual, and the order option in the pop-up menu affects the content of the layer.



- To select another layer, just **click** on it.
- To add a new layer, click on the “+” button.
- To delete the selected layer, click on the “x” button.
- To remove a layer from the view, click on the **Eye** icon of the layer: you will see all the objects of the layer disappear.
- To lock all the objects of a layer, click on the **Lock** icon of the layer: all the objects of the layer will become un-selectable.
- Drag and drop functions work very well in the layer toolbar: you can re-arrange the orders of the layer with the mouse.

Please note that the layers do not affect the Windows Controls objects (like the combo box object or the button object): they are always displayed in front of every other object.

## 6. The Objects

The Frame editor allows you to drop objects on the frame. It is time now to see the objects in details.

### The Backdrop object

The **Backdrop object** is used to draw **backgrounds** in the frame. It lies behind all the other objects, and does not interfere with the action. In fact it does nothing but sit there.

#### To create a Backdrop object

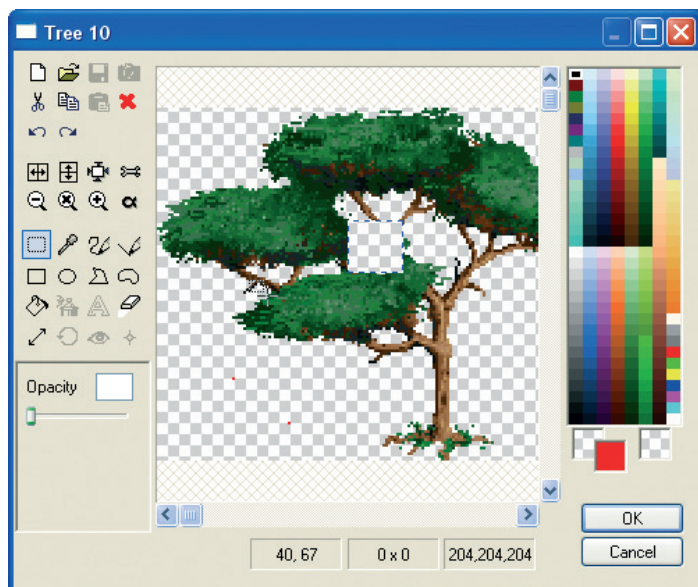
Choose the “**Insert object**” option in the frame editor pop-up menu, or in the Insert submenu in MMF main menu. Then choose the Backdrop object and drop it somewhere in the frame.

The icon of the object should appear: 

Of course, you can draw what you want in the object. To open the Multimedia Fusion **Picture editor**, right click on the backdrop object and choose the Edit option.

### 6.1. The Picture Editor

The picture editor is where you edit the pictures of the backdrop objects, and any other object with pictures.

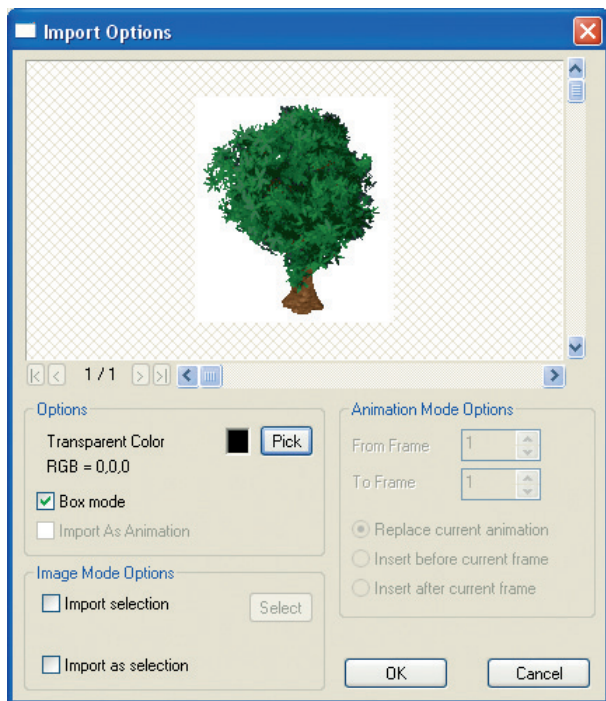


The Picture editor looks a lot like any other drawing program, and it does function like them. So we will only concentrate on some important icons. If you want the full detail of the icons, refer to the Help file in Multimedia Fusion.



## Import image

This icon opens a file selector for you to choose an image to import. Multimedia Fusion supports many kind of images: Targa, Png, Jpeg, Gif, Bmp, Avi and Pcx. Once the image selected, you are presented with the importation dialog box in which you can enter further parameters.



This dialog presents to you the image to be imported, and provides several options.

- **Transparent color.** The transparent color is a color that is not seen in the object : one can see what is behind the object when the transparent color is used. As a default, the color Black is the transparent color, but you can choose to pick another color from the picture: just click on the Pick button and then click somewhere in the picture. You can also double click on the colored square to choose a color.
- **Box mode.** Please refer to the Help file for information about this option. For normal importing, this option should be OFF.

- **Import selection.** Allows you to make a selection box around the graphics you want to import. Click on the Select button and then make the box with the mouse pointer in the picture.
- **Import as selection.** The image will be imported as a floating selection and will not replace the current image.

## Crop

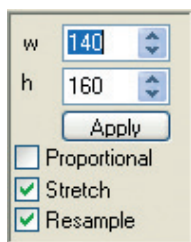
The crop button reduces the size of the picture intelligently: it removes the transparent areas around the image, thus keep only the useful part of the image. This option is very important to save memory in your application: only keep the important parts of the images to avoid wasting memory and disc space.

## Options

This icon allows you to choose the action of a right click in the image. You can two choices: draw with background color, or pick color (a click on a pixel will select its color for the left mouse button).

## Resize

This icon is used to resize the picture. When you click on this icon, a small menu appears in the information space under the icon:



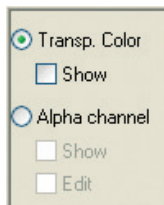
- Enter the new **width** and **height** in the **edit** boxes.
- **Proportional.** If you check this box, the width or height is automatically calculated from the other component to keep the proportionality of the picture intact during the resize.
- **Stretch.** The image will be resized as well. If not selected, blank areas are added around the image.
- **Resample.** Will resample the image during the process, leading to a smoother resize.

## Transparency

This tool allows you to show or hide the transparent color, and to edit it. In Multimedia Fusion this tool also allows you to display and edit the alpha channel of the image. If you click on this button, it opens the following box in the information zone:

- **Transp. Color.** Select this option to see the transparent color, click on **Show** to display the color.

- **Alpha channel.** Select this option to go in Alpha channel mode. The **Show** check box will display the alpha channel (the display will be replaced by the alpha channel), and a click on the **Edit** button permits to edit the picture: the color palette is replaced by shades of grey, and you have all the drawing tools available to draw the alpha channel.



## The properties of the Backdrop object

The backdrop object does not contain a lot of properties. But you will find in the Runtime options tab of the property toolbar an “Obstacle” property. This property is particularly important for platform games: if selected, the object will generate collisions with the moving objects.

The Obstacle property is a combo box with four choices:


- **None**: the object is not an obstacle
- **Obstacle**: the object is a classic obstacle
- **Platform**: for use with the platform movement, allow the player to walk on the object and go through it from bottom to top
- **Ladder**: defines the object as a ladder. The player will be able to climb on the object (to go from one platform to another).

## THE ACTIVE OBJECT

The Active object may be the most important object of Multimedia Fusion. It is a versatile and powerful object. As opposed to the backdrops objects, the active object takes an active part in the animation and movement in the frame: it can be animated, can move around the frame, generate collisions, be controlled by the joystick, store its own values, etc.

### To create An Active object

Choose the “**Insert object**” option in the frame editor pop-up menu, or in the Insert submenu in MMF main menu. Then choose the Active object and drop it somewhere in the frame.

The icon of the object should appear in the left pane of the frame editor: 

Of course, you can draw what you want in the object. To open the Multimedia Fusion **Animation editor**, right click on the active object and choose the Edit option.

## 6.2. The Animation Editor

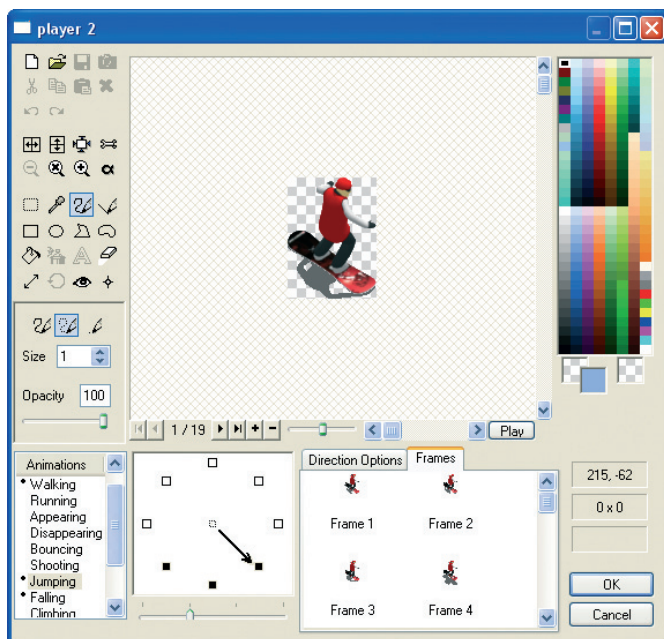
As its name indicates, the animation editor allows you to edit the animations of the active object. The active object contains twelve basic animations and you can define extra animations as well. The animations are the following:

- **Stopped**: displayed when the object is static
- **Walking**: displayed when the object moves at a slow pace
- **Running**: displayed when the object is moving at a fast pace (speed greater than 75)
- **Appearing**: displayed when the object is created

- **Disappearing**: displayed when the object is destroyed
- **Bouncing**: displayed when the object is bouncing
- **Shooting**: displayed when the object is shooting an object
- **Jumping**: for platform movement, displayed when the player is jumping
- **Falling**: for platform movement, displayed when the player is falling
- **Climbing**: for platform movement, displayed when the player is climbing a ladder
- **Crouch down**: for platform movement, displayed when the player is crouching
- **Stand up**: for platform movement, displayed when the player is un-crouching
- **User animations**: an unlimited number of animations that you can add to the object. You can name them as you want.

Each animation can have up to **thirty-two** directions. For example, you can define a walking animation for when the object goes to the right, top, bottom and left.

Why so many animations and directions? When you attach a movement to an object (see later), the object moves in a direction depending on the user input or the movement itself. Multimedia Fusion selects the appropriate animation and displays it as the object moves. It is automatic so you do not have to care about it thus saving edit time. This feature makes the active object suited for many uses, from a bouncing ball in a breakout game, to a walking and jumping character in a platform game. Active objects can also be used as clickable buttons in a multimedia presentation.



The previous picture shows the animation editor at work. As you can see it is based on the picture editor, so please refer to this part of the documentation for more information on how to use the drawing tools. We will concentrate on the new features of the animation editor. Looking at the bottom of the window, from left to right:

- **The animation selector.** Double click on an animation to open it. The different directions and frames will be displayed in the window. To add a user animation, right click in the animation selector and choose new. You can cut, copy and paste the animation as any other object in Multimedia Fusion.

- **The direction selector.** All the directions are displayed there. You can have up to 32 different animations for a single animation. To increase or reduce the number of animations, use the slider under the control, the different steps are 4 directions, 8 directions, 16 and 32. 32 is the finest, and you should use it for important objects, as it consumes the most memory. If you click with the right button on the filled dot of an existing direction, you open a pop-up menu...

- **Cut / Copy / Paste / Delete.** The usual clipboard functions: they all work with animation directions.

- **Create rotated directions.** This powerful option will create all the remaining directions from the selected ones by rotating the pictures. For example, if your current direction is right, all the other directions (top, left, bottom if you have 4 directions selected) will be created. The object is rotated around its hot spot (see later).

- **Create opposite direction.** Same as above, but the action only creates one new direction, the opposite.

- **Flip horizontally.** Will reverse all the frames of the animation horizontally.

- **Flip vertically.** Will reverse all the frames of the animation vertically.

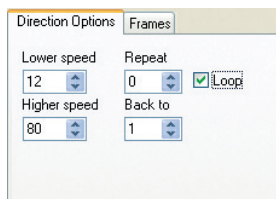
- **Invert frame order.** Will replace the first frame of the animation by the last one, the second by the previous etc.

|        |
|--------|
| Cut    |
| Copy   |
| Paste  |
| Delete |

|                           |
|---------------------------|
| Create rotated directions |
| Create opposite direction |

|                    |
|--------------------|
| Flip Horizontally  |
| Flip Vertically    |
| Invert Frame Order |

- **Direction options.** This tab present you with the different animation options, for each direction. It presents you with a set of controls for you to set:



- **Lower speed.** Sets the speed of the animation for when the object is static. The speed indicator can vary from 0 (stopped) to 100 (fast).

- **Higher speed.** Sets the speed for when the object is moving at maximum speed. For any intermediate speed, a calculation is done based on the low and high speeds, and the resultant speed is used.

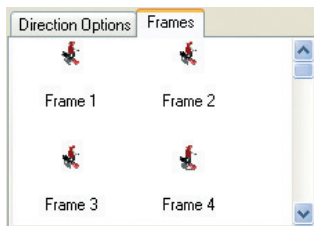
- **Repeat.** If you want your animation to be played a certain amount of times, enter a

number here. Using 1 causes your animation will play one time, 2 for two times, etc.

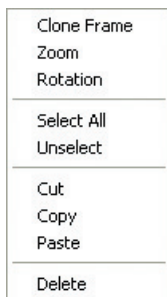
- **Loop.** Check this box if you want your animation to loop forever.

- **Back to.** You may not want your animation to loop to frame #1. You may want it to loop to frame 3 for example. Enter in this box the number of the frame you want.

- **Frames tab.** This tab displays in icons, the different frames of the selected animation and direction. A click on the icon opens the image in the picture editor, for you to modify it.



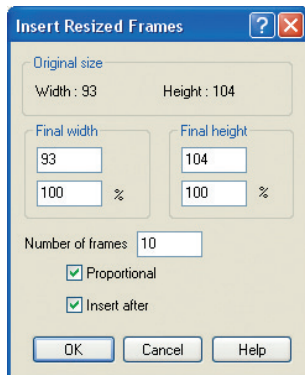
If you click with the right mouse button on a frame, it opens the following pop-up menu:



- **Clone frame.** This option will duplicate the current frame and place the new frame at the end of the frame list.

- **Zoom.** Opens a zoom dialog box:

This dialog box allows you to insert a number of frames calculated by zooming in or out the current picture.



- **Final width.** Enter the size of the object after zoom. You can enter a size in pixels or a percentage.

- **Final height.** Same for the height.

- **Number of frames.** Enter the number of frames you want the resize to be calculated on.

- **Proportional.** Makes the X and Y size be proportional: when you enter a size, the other is calculated automatically...

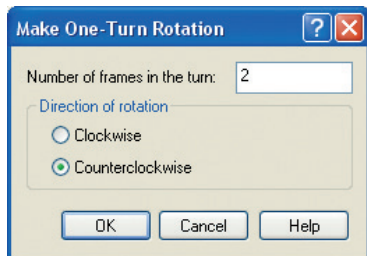
- **Insert after.** If not selected, the images will be added before the current image. If ticked, they will be added after.

- **Rotation**

Enter the number of frames for the rotation in the edit box, choose Clockwise or Counterclockwise and click on OK: the rotated frames will be inserted after the image you are editing.

- **Select all / Unselect.** Usual functions to select or deselect the images.

- **Cut / Copy / Paste / Delete.** Clipboard functions that work as usual.



- **The navigation bar**

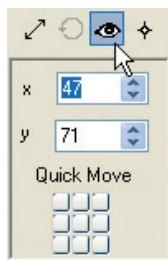
This bar allows you to navigate through the frames of an animation. The “+” button adds a



new frame in the animation, and the “-” button deletes the current frame. The slider is a very handy tool to create animations: it allows you to superimpose the previous or next frame to the current frame. You can easily draw the animation based on the previous frame.

- **The hot-spot**

The notion of hot-spot is important to master for animated objects. The hot-spot of an object is the point that corresponds to the on screen coordinates of the object. For example, the hot-spot of a ball or a race-car seen from above will be in the middle of the object, whereas the hot-spot of a walking character in a platform game will be between its feet. Hot-spots are also important for **rotations**: the rotations are calculated around the hot-spot, it is important to position the hot-spot properly before attempting a rotation.



To edit the hot-spot of a frame, click on the “**View hot-spot**” button in the animation editor. This will display a dot in the frame, and you can use the mouse to position it correctly. You can also enter the co-ordinates directly in the edit zones, or click on one of the pre-set positions available.

To set the hot-spot position in all the frames contained in the current direction, hold down the ALT key while positioning the hot-spot.

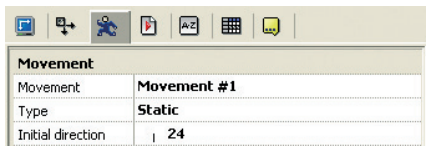
- **The action point**

The action point is similar to the hot-spot. When you do a “**Shoot**” action in the event editor, the bullet is created at the position of the action point: it is therefore important to position the action point properly before doing a Shoot action.

To position the action point click on the “**View action point**” icon in the animation editor.

### 6.3. The properties of the Active Object

The active object contains a lot of properties. The most interesting ones are the movement properties. Active objects are animated, but they can also move on the screen and create some action in your game. To add a movement to an object, select the object in the frame editor to display its properties, then click on the “**Movement**” tab in the property toolbar.



The first property named “**Movement**” allows you to define more than one movement for the object. For example, the first movement could be a path movement, and the second a bouncing ball. This property is reserved for advanced users.

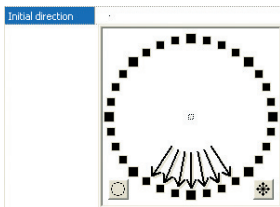
The **Type** property allows you to choose the type of movement for your object. Multimedia Fusion comes with eight different movements for you to choose from. Let’s see them in details.

## THE STATIC MOVEMENT

This is the simplest movement of all: the object does not move. This is the default movement when you create a new object in the frame editor. Even if the object does not move, it takes a part in all the collisions of the application: you can use the static movement for traps or triggers in a game, and all the things that do not need to move.

### Properties

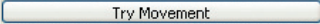

• **Initial Direction.** As the name indicates, this property sets the direction of the object when the application starts. Click on it and it opens a direction selector



This direction selector shows 32 possible directions. Highlight the ones you wish by clicking on the filled in square for that direction. Clicking a second time will remove that direction from use. If more than one direction is selected, a direction will be chosen at random between the selected ones.

## THE BOUNCING BALL MOVEMENT

This movement simulates the movement of a ball: the object goes in straight lines until it encounters an object where it can bounce, or otherwise react, on this object. This is the movement of choice for break-outs, pool, or other ball type games. The bouncing ball movement can also be used by advanced users in a variety of ways, as it is very flexible.

| Movement  |   |
|---|---|
| Movement  | <b>Movement #1</b>  |
| Type  | <b>Bouncing Ball</b>  |
| Initial direction                                   | 0,1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,<br> |
| <b>Speed</b>  |   |
| Speed   | 60   |
| Deceleration  | 0   |
| <input checked="" type="checkbox"/> Moving at start |   |
| <b>Bounce</b>                                       |   |
| # of angles   | 32  |
| Randomizer  | 20  |
| Security  | 60  |

- **Initial direction.** Same as above.
- **Try movement.** Click this button and you will see your object move accordingly to the settings in the properties. Useful for testing.
- **Speed.** The speed of the object, from 0 to 100. In Multimedia Fusion, all the speeds, accelerations and decelerations range from 0 (stopped) to 100 (fast). To make you an idea about the speed, use the “Try movement” button.
- **Deceleration.** Indicates that the ball will slow down gradually and finally stop. If 0 then the ball never stops. This can be thought of as friction.
- **Moving at start.** If checked, the object will move at the beginning of the application. If not, it will be stopped and you will have to start it with a “Start” action.
- **Number of angles.** You can choose between 8, 16 or 32 angles. The more angles, the finer the movement will be.
- **Randomiser.** This property adds a little bit of random in every bounce, making the ball bounce in unpredictable directions. 0 indicates no random whatsoever.
- **Security.** When a ball get stuck in bounces (it does always the same pattern), the security property is important: it automatically unblocks the object after a certain amount of bounces.

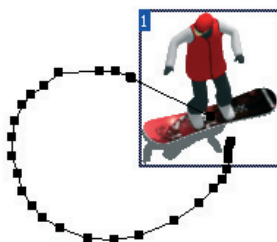
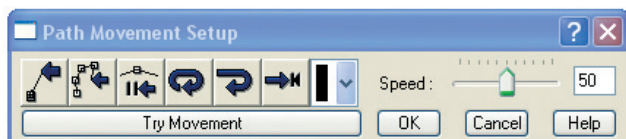
## THE PATH MOVEMENT

The path movement is a very powerful movement that allows you to record the mouse and draw a path that your object will follow during runtime. You actually draw the path on the screen with the mouse.

| Movement      |                                     |
|---------------|-------------------------------------|
| Movement      | <b>Movement #1</b>                  |
| Type          | <b>Path</b>                         |
| Edit movement | <input type="button" value="Edit"/> |

As you can see, the properties of the path movement are limited to a simple “Edit” button. A click on this button opens the path movement editor.

**The buttons, from left to right**



- **New line.** Draws a single line with the mouse. If you click with the right mouse button to end the line, a new line will be automatically inserted.
- **Tape mouse.** Records the mouse movement: click with the left mouse button and draw the path you wish. The recording ends when you release the mouse.
- **Set a pause.** To use this button, you must first select a node in the path. In the dialog box, you can enter a time: the object will wait for that amount of time when playing back the path.
- **Loop the movement.** Makes the movement restart when it reaches the end of the path. Warning, the object is not repositioned at the original location, so it will continue on its path.
- **Reverse at end.** If selected, the path movement will reverse when the end is reached, in a Ping-Pong effect. When it reaches the first node, and if the “Loop movement” button is set, it will restart forward.

- **Reposition at end.** This option will enforce the position of the object to the original location when the movement end.
- **Color selection.** Choose the color you want to display the path.
- **Speed.** You must first select one or more nodes to use this control. Sets the speed of the current node, from 0 (stopped) to 100 (maximum speed).
- **Try movement.** Click to try the movement and immediately see the results.

You can with the mouse, select one or more nodes, move them around, and delete them with the delete key. A click with the right mouse button on one of the nodes opens the following menu:

|   |
|---|
| Draw line<br>Record mouse<br>Set pause<br>Set speed<br>Set name |
| Delete  |

- Draw line. Same as the “Draw line” button.
- Record mouse. Same as the “Tape mouse” button.
- Set pause. Same as the “Set a pause” button.
- Set speed. Allows you to enter the speed of that node.
- Set name. Allows you to give a name to this node. Names can be important for advanced users: you can, in the programming of events, detect when a node is reached and take action accordingly.

## THE PINBALL MOVEMENT

The Pinball movement is a bouncing ball movement with gravity. The ball bounces like a pinball when colliding with obstacles and on the floor. This movement is only present in Multimedia Fusion.

- **Gravity.** Indicates the gravity factor, from 0 (no gravity) to 100 (strong gravity). The gravity affects the shape of the bounces and the height of the bounces.
- **Deceleration.** Indicates a deceleration factor that will stop the ball after a while. From 0 (no deceleration) to 100 (immediate stop).
- **Move at start.** If checked the object will move when the frame starts.
- **Initial speed.** Indicates the initial speed of the object, if it is to move at start.
- **Initial direction.** Contains the initial direction of the object when the frame starts.

| Movement  |  |
|---|--|
| Movement  | Movement #1                            |
| Type  | Pinball movement                       |
|   | Try Movement                           |
| Gravity   | 25                                     |
| Deceleration                                      | 0                                      |
| <input checked="" type="checkbox"/> Move at start |  |
| Initial speed                                     | 50                                     |
| Initial direction                                 | 0,1,2,3,4,5,6,7,8,9,10,11,12,13,14,15, |

## THE MOUSE CONTROLLED MOVEMENT

This movement is controlled by the user, with the mouse. Basically the object moves accordingly to the movements of the mouse, like the cursor.

| Movement      |                                     |
|---------------|-------------------------------------|
| Movement      | <b>Movement #1</b>                  |
| Type          | <b>Mouse Controlled</b>             |
| Player        | <b>1</b>                            |
| Edit movement | <input type="button" value="Edit"/> |

**Player.** Indicates which player controls the movement, can be from 1 to 4.

**Edit.** Click on this button to edit the mouse controlled movement. It opens the following box on the frame:



The zone drawn with handles on the frame represents the zone of movement allowed for the object. Drag the handles with the mouse pointer to enlarge / reduce the zone. Click on "Try movement" to see the movement in action. Please note that the mouse pointer disappears when the mouse movement is running, you have to press "**ESCAPE**" to quit the movement demo.

## THE EIGHT DIRECTIONS MOVEMENT

This movement is also controlled by a player, using a joystick or the arrow keys. It is a simple movement: the object goes in the direction wanted by the player. A total of 8 directions are possible (counting diagonals).

| Movement  |   |
|---|---|
| Movement  | <b>Movement #1</b>                          |
| Type  | <b>Eight Directions</b>                     |
| Player  | <b>1</b>                                    |
| Directions  | * 0,4,8,12,16,20,24,28                      |
| Initial direction                                   | * 0,4,8,12,16,20,24,28                      |
|   | <input type="button" value="Try Movement"/> |
| <b>Speed</b>  |   |
| Speed   | <b>50</b>                                   |
| Deceleration  | <b>50</b>                                   |
| Acceleration  | <b>50</b>                                   |
| <input checked="" type="checkbox"/> Moving at start |   |

- **Player.** Choice of the player who controls the movement, from 1 to 4.
- **Directions.** Click on this property and select the possible directions out of the 8 possible. If a direction is unselected, then the object will not be able to follow it, even if the player presses the correct key on the keyboard.
- **Initial direction.** Click and choose the initial direction of the object, out of 8.
- **Try movement.** Click to see the movement in action.

- **Speed.** Maximum speed of the object, from 0 to 100.
- **Acceleration.** Acceleration factor, from 0 (no acceleration) to 100 (immediate).
- **Deceleration.** Deceleration factor, from 0 (no deceleration) to 100 (immediate). The deceleration factor indicates the time taken to go from the maximum speed to stopped when the user releases the keys.

## THE RACE CAR MOVEMENT

As its name indicates, the race car movement makes the object run like a car, seen from above. With the keys, you can turn right or left, and move forward or backward in the current direction of the car.

**Note:** as the movement uses the 32 possible directions, it is best seen with an object in which all the 32 directions of the Stopped or Walking animations are defined.

- **Player.** Number of the player who controls the object, from 1 to 4.

- **Initial direction.** Choose the initial direction of the object with this property.

- **Try movement.** Click to see the movement in action.

- **Speed.** Maximum speed of the object, from 0 to 100.

- **Acceleration.** From 0 to 100: time taken to go from stopped to maximum speed.


- **Deceleration.** From 0 to 100: time taken to go from maximum speed to stopped.

- **Enable reverse.** If checked, the car will be able to go backward when the user presses the bottom key of the joystick or the down arrow key on the keyboard.

- **Moving at start.** If checked makes the car move when the frame begins.

- **Number of angles.** Choose here the number of possible angles, between 4, 8, 16 or 32. The more angles, the finest the rotation effect.

- **Rotating speed.** From 0 to 100: indicates the speed of rotation of the car when the user presses left or right keys.

| Movement  |   |
|---|---|
| Movement  | <b>Movement #1</b>  |
| Type  | <b>Race Car</b>   |
| Player  | <b>1</b>  |
| Initial direction                                   |  <b>0,1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,</b> |
|   | <a href="#">Try Movement</a>  |
| <b>Speed</b>  |   |
| Speed   | <b>50</b>   |
| Acceleration  | <b>50</b>   |
| Deceleration  | <b>50</b>   |
| <input checked="" type="checkbox"/> Enable reverse  |   |
| <input checked="" type="checkbox"/> Moving at start |   |
| <b>Rotation</b>                                     |   |
| # of angles   | <b>32</b>   |
| Rotating speed                                      | <b>25</b>   |

# THE PLATFORM MOVEMENT

The platform movement makes the object move like a character of a platform game: it can walk on the left and right, jump, climb on ladders and platforms, and fall from one platform to another.

In order to work, the frame must contain backdrop objects (or quick backdrop) and these objects must have the "Obstacle", "Platform" or "Ladder" property set. You must also define an event in the event editor that stops the object when it collides with the background.

This movement works best with an object in which all the animations are defined.

| Movement  |                    |
|---|--------------------|
| Movement  | <b>Movement #1</b> |
| Type  | <b>Platform</b>    |
| Player  | <b>1</b>           |
| Initial direction                                   | <b>+ 0,8,16,24</b> |
| <div>Try Movement</div>                             |                    |
| <b>Speed</b>  |                    |
| Speed   | <b>50</b>          |
| Acceleration  | <b>50</b>          |
| Deceleration  | <b>50</b>          |
| <input checked="" type="checkbox"/> Moving at start |                    |
| <b>Jump</b>   |                    |
| Gravity   | <b>50</b>          |
| Strength  | <b>50</b>          |
| Control   | <b>Button 1</b>    |

- **Player.** Choice of the player who controls the movement, from 1 to 4.

- **Initial direction.** Choose the initial direction of the object.

- **Try movement.** Click to see the movement in action.

- **Speed.** Maximum speed of the character, from 0 to 100.

- **Acceleration.** Time taken to go from stopped to maximum speed, from 0 to 100.

- **Deceleration.** Time taken to go from maximum speed to stopped, from 0 to 100.

- **Moving at start.** Check this box and the object will move at the start of the frame.

- **Gravity.** Strength of the gravity, from 0 to 100. A strong gravity will reduce the jumps and increase the speed of falling.

- **Strength.** Strength of the jumps. This property should be set considering the gravity factor: the higher the gravity, the stronger the strength of the jumps have to be.

- **Control.** Indicates the combination of keys necessary to initiate a jump. Can be nothing (no jump), fire button 1 or 2, or a combination of the left and right keys plus the up key.

## Other properties of the active object

The active object contains a lot of different properties, for an exhaustive description, please refer to the electronic help file.

| Values                   |                           |
|--------------------------|---------------------------|
| <b>Alterable Values</b>  |                           |
| Alterable Value A        | <b>0</b>                  |
| Second value             | <b>1000</b>               |
| <div>New</div>           |                           |
| <b>Alterable Strings</b> |                           |
| Alterable String A       | <b>"This is a string"</b> |
| <div>New</div>           |                           |

### • The Alterable Values

The alterable values of the active object are internal memory locations in which you can store numbers. You can have up to 26 different alterable values and you can name them as you want. To edit the names of the alterable values and set their content for when the frame starts, click on the “Values” tab in the property toolbar. Click on the “New” button to add a value, click on the name of the value to edit it, and click on the number to change it.

### • The Alterable Strings

Alterable strings are identical to the alterable values, but they contain strings of text. You can have up to 10 different alterable strings in an active object. They are accessible with the same “Values” tab in the property toolbar.

## 123 THE SCORE OBJECT

The score object is a game oriented object: it displays the current score of a player. The event editor contains actions to modify the score of a player. The score object reflects these modifications.

### To create a Score object

Choose the “Insert object” option in the frame editor pop-up menu, or in the Insert submenu in MMF main menu. Then choose the Score object and drop it somewhere in the frame.

The icon of the object should appear in the left pane of the frame editor:

### The properties

The score object does not contain many properties.

| Settings |                                     |
|----------|-------------------------------------|
| Player   | 1                                   |
| Type     | Text                                |
| Images   | <input type="button" value="Edit"/> |


- **Player.** Indicates which player the object is displaying the score, from 1 to 4.
- **Type.** Use this property to set the display of the object. It can be “Text”, in this case the object will be displayed using a font (you can change this font in the Text tab of the property toolbar). It can also be “Numbers”: click on the Edit button to open the picture editor and change the images.

## THE LIVES OBJECT

The lives object is also a game oriented object: it displays the current number of lives of the player. The event editor contains actions to modify the number of lives of a player. The lives object reflects these modifications.

### To create a Lives object

Choose the “Insert object” option in the frame editor pop-up menu, or in the Insert submenu in MMF main menu. Then choose the Lives object and drop it somewhere in the frame.

The icon of the object should appear in the left pane of the frame editor: 

### The properties

This object also does not contain a lot of properties.

| Settings |                                     |
|----------|-------------------------------------|
| Player   | 1                                   |
| Type     | Image                               |
| Image(s) | <input type="button" value="Edit"/> |


- **Player.** Indicates the player whose number of lives is to be displayed. From 1 to 4.
- **Type.** Use this property to set the display of the object. This combo box can have three values. “Image”: the object will show the number of lives as the repetition of a graphic. Click on the Edit button to open the picture editor and modify the image. “Text”: the number of lives is shown as a number, printed in the selected font. You can change this font by opening the Text tab of the property toolbar. Finally, “Numbers” will display the number of lives with digits. You can edit the digit by clicking on the Edit button.

## THE COUNTER OBJECT

The counter object is as its name indicates a counter. You can display numbers and perform calculations with it.

### To create a Counter object

Choose the “Insert object” option in the frame editor pop-up menu, or in the Insert submenu in MMF main menu. Then choose the Counter object and drop it somewhere in the frame.

The icon of the object should appear in the left pane of the frame editor: 

## Properties of the counter object

When you click on a counter object in the frame editor, its properties are displayed in the property toolbar:

| Settings      |            |
|---------------|------------|
| Value         |            |
| Initial Value | 0          |
| Minimum Value | -999999999 |
| Maximum Value | 999999999  |
| Display       |            |
| Type          | Text       |

- **Initial value.** Enter the initial value of the counter: the counter will contain this value when the frame starts. The default value is 0.
- **Minimum value.** Defines the lowest possible value contained by the counter.
- **Maximum value.** Defines the highest possible value contained by the counter. Note: you choose to display your counter as a horizontal or vertical bar, you must enter correct values in these two fields, otherwise the bar will not be displayed properly.
- **Display type.** This combo box proposes several choices on how to display your counter.
- **Text.** The counter is displayed as a string of text, using the chosen font. You can choose the font in the "Text" tab of the property toolbar.
- **Hidden.** The counter is not shown. But you can still use it in the event editor to perform calculations and retain numbers.
- **Numbers.** The counter is displayed with graphical numbers. A click on the Edit button will open the picture editor with all the digits ready to be modified. To import a font directly, select all the frames, and click on the Import font tool of the picture editor. Choose a new font and click on OK, the font is imported.
- **Vertical bar.** This option makes the counter displayed as a vertical progress bar.  
Count indicates the direction of the bar, up or down. Fill type can be Solid color or Gradient, just like the Quick backdrop object. Color allows you to choose the color of the bar in the palette.
- **Horizontal bar.** This option makes the counter displayed as a horizontal progress bar. Count indicates the direction of the progression bar, either From right or From Left. Fill type can here too be chosen between Solid color or gradient.
- **Animation.** Displays the counter as images. The value gives the number of the image to display, based on the minimum and maximum values. Click on the Edit button to open the picture editor and edit the images.

|           |                    |
|-----------|--------------------|
| Display   |                    |
| Type      | Vertical bar       |
| Count     | Up                 |
| Fill type | Solid color        |
| Color     | RGB = 60, 179, 113 |

|  |                    |
|--|--------------------|
| Display                                    |                    |
| Type                                       | Horizontal bar     |
| Count                                      | From right         |
| Fill type                                  | Gradient           |
| Color                                      | RGB = 60, 179, 113 |
| Color 2                                    | RGB = 255, 0, 0    |
| <input type="checkbox"/> Vertical Gradient |                    |

# THE STRING OBJECT

The String object contains a string of text. It can display the string using a selected font. It is the object to use as soon as you have some text to display.

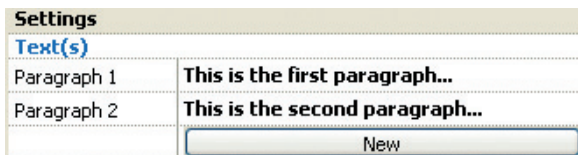
## To create a String object

Choose the “**Insert object**” option in the frame editor pop-up menu, or in the Insert submenu in MMF main menu. Then choose the String object and drop it somewhere in the frame.

The icon of the object should appear in the left pane of the frame editor: 

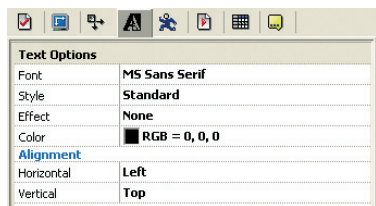
## Properties

Only a few specific properties for the string object.



The string object can contains a number of **paragraphs**. By clicking on the “**New**” button you add a new paragraph to the list. Click on the text zone to edit the text of the paragraph. In the event editor, you will be able to display this or that paragraph.

To change the font used to display the object, click on the “**Text Options**” tab in the property toolbar. This will display the font choice and text formatting option:



- **Font.** Click to choose another font.
- **Style.** Allows you to display the text in bold or italic.
- **Effect.** Choice of underline or strike-out.
- **Color.** Choose the color of the font.
- **Alignment / Horizontal.** To align on the left, center or right justify the text.
- **Alignment / Vertical.** To display the text in the top, middle or bottom of the object's box.

The string object also has an option called Alternate String which can be accessed from the Event Editor. You can have the string change its text during runtime to anything you might require. The string object is also an invaluable object to store strings, retrieve them and make string calculations in the event-editor.

## 6.4. Object inventory

In this chapter we will describe several of the remaining objects briefly to give you a quick view of all the possibilities contained in Multimedia Fusion. If you want to discover more about an object, you will find its documentation in the help file. Or you can just drop the object in the frame, inspect its properties, conditions and actions: most objects are self-explanatory.

### Audio objects

- **CDAudio object.** Use this object to play the audio tracks of a CD inserted in the CD drive of your computer.
- **MCI object.** Using Windows' MCI interface, you can play sounds and music.
- **Mixer object.** Change the volume of the sounds and music on your computer.

### Control objects

- **Clickblocker object.** Block the clicks of the mouse with this object. *Only in MMF.*
- **Analog joystick.** Allow you to test the position of an analogue joystick plugged on the computer.

### Data objects

- **Array object.** Store information in an array that you can share between frames.
- **Counter object.** Store and display numbers.
- **Datagrid object.** Display data in a grid like presentation, like an Excel sheet. *This object is only available in Multimedia Fusion Developer.*
- **Search object.** Search for strings of text in files and directories. *Only in MMF.*
- **SharedData object.** Share numbers and strings between two or more Multimedia Fusion applications running on the same computer.

### Database object

- **ODBC object.** Access any database using the simple syntax of the ODBC object. *Only in MMF.*

### File objects

- **File object.** Manipulate, copy, delete files located on your hard disc or floppies.

### Game objects

- **HiScore object.** Automatically manage a hi-score table for your games.
- **Lives object.** Displays the number of lives of a player.
- **Score object.** Displays the score of a player.

### Graphic and animation objects

- **Active object.** Powerful all-purpose object for games and serious applications.

- **Backdrop object.** Let you define backdrops in your application.
- **Quick Backdrop object.** Allows you to draw rectangles and ellipses in the back of your frame. Like the Backdrop object, contains the “Obstacle” property.
- **Active picture object.** Loads images from disc and display them in your frame. This object can resize and rotate the images.
- **Animation object.** Display FLI, GIF and files with this object.
- **Draw object.** Draw directly on the screen using the mouse. **Only in MMF.**
- **Picture object.** Loads images from disc and display them as background.
- **Screen capture object.** Capture part or the entire screen with this object. **Only in MMF.**

## Interface objects

- **Button object.** Display pushbuttons, radio buttons and check boxes in your application with this object.
- **Combo box object.** Create combo boxes and handle them.
- **Cursor object.** Change the shape of the mouse pointer to your heart’s desire. **Only in MMF.**
- **Dialog box object.** This powerful object allows you to create dialog boxes and manage them in your application. **This object is only available in Multimedia Fusion Developer.**
- **Edit object.** Open text edit zones on the frame, useful when asking the user for text information.
- **Explorer object.** Open explorer windows in your application, to display the content of disc drives. **This object is only available in Multimedia Fusion Developer.**
- **List object.** Open list boxes in the frame of your application much like the Combo Box object.
- **Listview object.** Displays and manage a listview control, Microsoft way of displaying lists of items. **This object is only available in Multimedia Fusion Developer.**
- **Pop-up menu object.** Create and display pop-up menus in your application. **Only in MMF.**
- **Window shape object.** Change the shape of the application window to whatever you like and create animated items floating on the desktop.
- **Static text object.** Display non-interactive text with this control.
- **Sub-application object.** Start and run MMF applications within your main application.
- **System box object.** Create graphical buttons and toolbars with this object. **Only in MMF.**
- **Tree control object.** Display directories, drives and data in a tree-like structure. **This object is only available in Multimedia Fusion Developer.**
- **Window control object.** Control the size, position of the application window with this object.

## Internet objects

- **FTP object.** Download and upload files from the Internet with this object. **Only in MMF.**
- **Vitalize! Plugin object.** Control the Vitalize! Plugin from this object. Vitalize! Enables

you to display your creations in an Internet browser.

## Math objects

- **Double precision object.** Calculate in double precision with the functions of this object.

Only in Multimedia Fusion Developer.

## Network objects

- **Network object.** Communicate between two distant machines using this object.

Only in MMF.

## Printer objects

- **Print object.** Print images and text with this object.

## Storage objects

- **INI object.** Create, read and manage INI files.
- **Array object.** Store array data on disc.

## System objects

- **OS object.** Retrieve information about the machine, the version of Windows and the environment variables. This object is only available in Multimedia Fusion Developer.

## Text objects

- **Question and answers object.** Create quiz with this object who displays the questions and the answers automatically.
- **Rich edit object.** Display and edit complex text with this object, and create your own version of Wordpad. Only in MMF.
- **Formatted text object.** Displays Rich Text Format texts in a graphical environment.
- **String object.** Display and retrieve simple strings of text.

## Date & Time objects

- **Date & Time object.** Display a clock, a calendar, or a countdown timer with this object.

## Video objects

- **AVI object.** Display video files in the Microsoft format AVI.
  - **DirectShow object.** Play sounds and video files or DVD using this powerful object.
- Only in MMF.
- **Quicktime object.** Play and manage the popular Quicktime files.

## Other objects

- **ActiveX object.** Use and drive ActiveX controls with this object. Only in MMF.
- **Layer object.** Manage the display priority between the different objects of your application.

## 7. The Event Editor

Shown as an array, the **event editor** is a grid in which you are going to display, create and modify all the events that will bring your frame to life.

In the frame editor, you can drop objects and set-up the display of your application. But if you try to run an application immediately after finishing your work in the frame editor, you will notice that very little happens. The moving objects move, and finish by getting out of the display, but nothing reacts to the mouse and the application just stays there doing nothing.

The event-editor is the place where you will design and create the logic behind your application. You will be able to detect the different states of the application (collisions, timer, position of the mouse etc.) and take actions to react.

### What is an event?

The event-editor is based on events. Events are what make Multimedia Fusion so unique: they are a simple yet powerful way of programming. Do not be afraid, there is no programming language in Multimedia Fusion, everything is done with the mouse by point and click.

So what is an event? It is the association of one or more **conditions** and one or more **actions**. Conditions and actions? They reflect your natural way of thinking. Imagine you have created a breakout game in the frame editor, and you want the ball to bounce when it collides with a brick. You would think about this as:

When the ball collides with the brick, make the ball bounce.

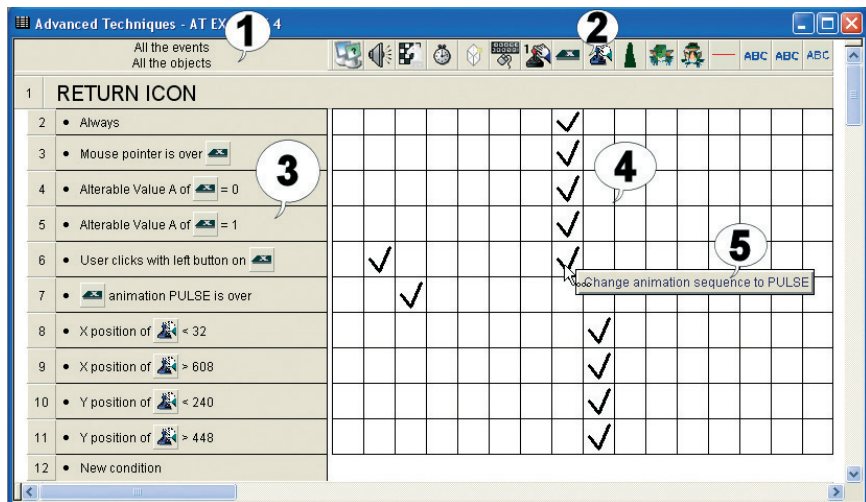
The previous line is an event. The condition is “When the ball collides with the brick”, and the action is “Make the ball bounce”. In the event editor, you would create a condition that detects the collision between the ball and the brick, then in the pop-up condition menus of the ball set the action to bounce the ball.

### How to reach the event editor

There are multiple ways of opening the event editor.

- In the **workspace toolbar**, right click on a frame, and choose event-editor
- In the **storyboard editor**, right click on a frame and choose event-editor
- Click on the **event-editor button** in the navigation toolbar
- Press CONTROL-E

# The display of the event-editor



The object bar displays the objects of the application. The objects dropped in the frame in the frame editor but also new objects, the system objects, that you will only find in the event editor. From left to right, let's see the system objects:



**The special object.** This object contains all the system conditions and actions. You can modify global values, perform comparisons, generate random numbers and mathematical functions.



**The speaker object.** It contains actions to play sounds and music, and to test if the sounds are played at a certain moment of time.



**The storyboard control object.** Contains all the necessary to control the flow of the program: going from one frame to another etc.



**The timer object.** Contains conditions to test for certain duration.



**The create object.** Allows you to create new objects in your application.



**The mouse and keyboard object.** Allows you test for a key depressed, the position of the mouse, and eventual clicks on the frame.



**The player 1 object.** Contains actions to change the score and number of lives of the player 1. Note that if more than one player is defined, you will find in the event editor icons for player 2, player 3 or 4.

If you click with the right mouse button on one of the objects, it opens the following pop-up menu:

|                             |  |
|-----------------------------|--|
| Hide                        |  |
| Insert hidden object        |  |
| Insert object folder        |  |
| <hr/>                       |  |
| Replace by another object   |  |
| Delete any action in column |  |
| <hr/>                       |  |
| Open behavior               |  |
| Hide objects with qualifier |  |

- **Hide.** Hides the object from the display. This option becomes handy if you have many objects displayed in the object bar. You can choose to hide them to limit the number of objects displayed.
- **Insert hidden object.** The counterpart of the previous option, allows you to pick one object in the hidden object list and to redisplay it.
- **Insert object folder.** Creates a folder in which you can drag & drop other objects. This folder can be closed thus saving a lot of space in the display.
- **Replace by another object.** Will replace all the actions and condition related to the first object by the second object.
- **Delete any action in column.** As its name indicates, this action will remove all the actions in the column. Very handy for example to destroy all the sound actions and make game silent in a click.

- **Open behavior.** Please refer to appendix 1 for more information on behaviors.
- **Hide object with qualifier.** The option is only available if you click on a qualifier. Will hide all the objects containing this qualifier.

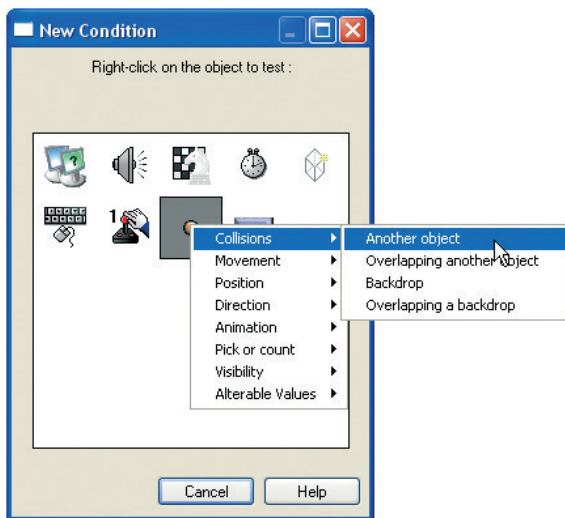
If you click with the left mouse button on one object, it **filters the events**: only the events containing this object will be displayed, thus reducing dramatically the number of lines displayed. Clicking with the left mouse button a second time on the object being filter will redisplay all the objects.

## How to add a new event?

It is very simple to add a new event. At first, the event-editor is empty, so the display should look more like this:



To define a new event, follow the simple procedure:



- Click on the **“New condition”** line. The last line of conditions will always contain the New condition option.
- A dialog box opens presenting to you all the objects of the application. Click with the right mouse button on the object you want to test. Let's imagine we want to test the ball object (called Ballgolden) for collision with the brick object (called Violet).

So we open the condition pop-up menu, find the collision sub menu and choose “Collision with another object”

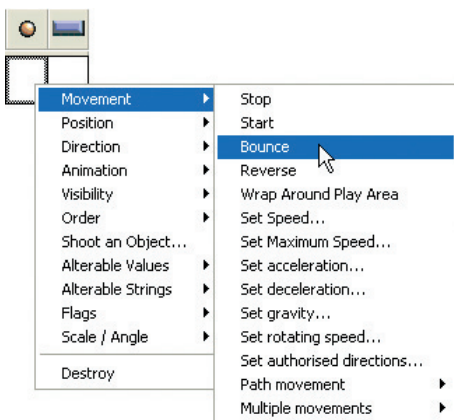
- A new dialog box immediately opens, asking for the object to test the collision with. Double click on the brick “Violet”
- The New condition is replaced with the condition we have just defined.



As you can see, the objects are displayed with their icons, but you can obtain the name of the object by moving the mouse pointer above the icons.

We have defined a **condition** but we need now the **action** to complete the event. Here the action we want to define is “Bounce the ball”.

- Locate the icon of the ball in the object bar, and go down in the empty square **below** the object, on the **same** line as our newly created “collision” condition.
- Click with the right mouse button to open a pop-up menu: the ball object’s action menu. In this menu locate the Movement submenu and in it the Bounce action:



- Release the mouse to close the menu, and you will see a new checkmark in front of the condition: our new event is defined.

As you can see, using the event-editor is really simple: it is a matter of choosing options in pop-up menus. Maybe you would think that this system is somewhat limited? Nothing is further from the truth:

- The number of conditions and actions is **very** high, we have added conditions and actions to cope with every aspect of an application.

- You can if you want, have **more than one** condition in an event, again called a compound condition. The actions will be executed when **all** the conditions are satisfied at the same time.
- You can of course have more than one action for one event: the actions are executed one after the other at high speed.

## Using the event editor

If you click with the right mouse button on a previously created condition, it opens the following pop-up menu:

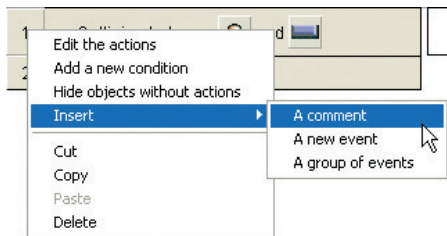
- **Edit.** Edits the current condition. For example, for a collision condition re-opens the box in which you can choose the objects.
- **Replace.** Replaces the current condition by a new one.
- **Insert.** Inserts a new condition after the current one. Useful to make complex events with multiple conditions called compound conditions.
- **Negate.** Negates the current condition making it detect the contrary of its default behavior. For example, the condition Object Is Colliding with another object negated will be true if the object is NOT colliding with the other object. Negations are marked by a big X before the condition.
- **Cut / Copy / Paste / Delete:** the usual edit functions. You can perfectly cut and copy conditions in the clipboard.

Edit  
Replace  
Insert  
Negate

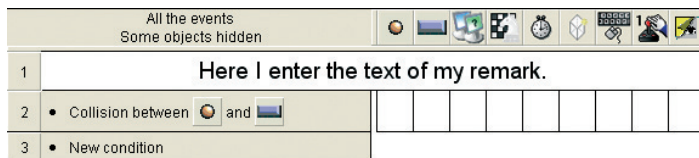
Cut  
Copy  
Paste  
Delete

If you click with the right mouse button on the number located on the left of the events, you get the following pop-up menu:

- **Edit the actions.** Opens the action editor: the action editor is a special version of the event-list editor (see later) that displays only the actions of this event. You can add and edit the actions as you wish, and more important you can set the order of the actions: ordering the actions may be necessary for your application as sometimes some actions rely on the previous actions to be executed.
- **Add a new condition.** As the name indicates, add a new condition to the event.
- **Hide objects without action.** This action will hide all the objects that do not contain any action from the display.
- **Insert / A comment.** Allows you to comment your code. Comments are pieces of text intermixed with the events that describe the logic of the application. You should as much as possible comment your code to make it more readable. When you choose this option, the comment-entry box is opened.



Type the text of your comment in the edit zone, choose the font with the “**Choose font**” button, the colors with the appropriate buttons, and click on OK. Here s how this comment is displayed in the event editor:

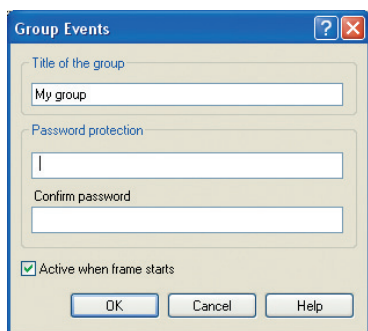


As you can see comments are very important to make a code clearer. Do not worry, comments do not take any time during the execution of the application as they are totally skipped by the runtime engine. Comments are extremely useful if multiple developers are sharing files. (Also many Clickteam developers use the background color as an indication of the complexity of a specific event.)

- **Insert / A new event.** Will open the object condition dialog for you to choose a new condition, and automatically create a new event with this condition.

- **Insert / A group of events.** Group of events can be a very handy way to simplify and organise your code. Basically a group is a set of events that have some level of commonality and are grouped together. Groups can be deactivated, in this case ALL the events within the group are ignored. You can use an action to reactivate the entire group all at once. Furthermore, groups can be closed by double clicking on them: it instantly hides all the events in a single line (the title of the group).

To create a new group, you have to enter its name in the group definition dialog:



Enter the title of your group in the proper field. You can decide to protect the group with a **password**: enter the password in the two password fields. Your group will be able to be closed, but to open it you would need the password. Use this option if you distribute your code for others to look at but do not want other people's eyes to peek inside this particular group of events. Warning, if you loose your password, there is no way to open the groups! So you should restrain from password-protecting your code until the end of the development.

**Active when frame starts:** if this box is checked, the group will be activated when the frame starts, if not, it will be inactivated. Deactivated groups are greyed out in the event editor.

- **Cut / Copy / Paste / Delete:** The clipboard functions are available for events.

## Drag & drop in the event editor

Using the event editor, you find soon find out that drag & drop is an indispensable part of the development process. You can drag:

- Conditions on one another, or from one event to another.
- Actions checkmarks from one column to another.
- Events on another events, like this you can re-order the events in your program.
- Closed or opened groups of events
- You can of course select more than one event to drag at the same time.

When you drag the elements, you will notice the action of the SHIFT and CONTROL keys. SHIFT deletes the origin of the drag. CONTROL copies (inserts) the drag at the new position.

## Conditions of interest

We will not review in this manual all the available conditions and actions of Multimedia Fusion: this would be way too long for a printed manual. We are just going to detail some of the most important ones, object per object. For an exhaustive list and description, please refer to the electronic documentation. Of course the best way to learn about an individual objects events is to experiment.



### Conditions in the special object

The special object contains several interesting conditions.

- **Always.** This condition is always true, no test is done. Useful for use within groups or for testing purpose.
- **Never.** This condition is always false. Very handy if you want to deactivate a line of event: simply insert a Never condition at the beginning.
- **Compare to a global value / string.** Global values and strings are a handy way to transmit data between frames. With this condition you can test their value.
- **Compare two general values.** Do not confuse with the previous condition. This condition enables you to make a comparison between two values, numerical or text.
- **Limit conditions / Run this event once.** If you insert this condition in an event, the event will only be run once in the duration of the frame.
- **Limit conditions / Only one action when event loops.** If the event is true, it will be true only once and remain false until the condition making it true is false again. Like a one shot event: executed only one cycle, then ignored until the conditions are false then true again.

- **Limit conditions / Repeat.** This condition will make the event repeat a certain amount of times. You are asked to enter a number when this condition is selected.
- **Group of events / Check for activation.** This condition will be true if the selected group of event is activated, or if negated (see above) if the group is not active.
- **Group of event / On start of group.** This condition will be true only if it used within a group, and if this group has just been activated. It will be true only one time, and after will become false until the group is deactivated and reactivated again.
- **On Loop.** Allows you to create fast loops: fast loops are a very powerful way of executing jobs very fast. More info in appendix 3.

## Conditions in the storyboard object



The conditions of the storyboard object test the flow of the application.

- **Start of frame.** This condition is only true once at the very beginning of the frame. It is executed before all the other conditions. You should perform initialisations (setting counter, strings, ect...) in the events that start by "Start of frame".
- **End of frame.** This condition is the last to be true when the frame ends.
- **End of application.** This condition is only true if the user quits the application. It is executed after the End of frame condition.

## Conditions in the mouse pointer and keyboard object



The conditions in this object allow you to test the keyboard and mouse.

- **Upon pressing a key.** This condition asks you to press the key to test. It is true once only when you have pressed the key.
- **Repeat while key is depressed.** This condition is similar to the previous one, but it will be true as long as you keep pressing the key.
- **Upon pressing any key.** This condition will be true if the user presses any key on the keyboard. Useful for menus or text readers.
- **Checks for mouse pointer in a zone.** Asks you to define a zone on the frame. During runtime, this condition is true when the mouse lies within the defined zone.
- **Checks for mouse pointer over an object.** This condition is true if the mouse pointer lies over the given object.
- **Checks for mouse clicked.** This condition is true if the mouse is clicked. The condition can be set for left, middle or right buttons and single or double clicks.



## Conditions in the timer object

- **Every.** This condition asks for a period of time, example 1 second. During runtime, it will be true every second in our example. Useful for flashing objects, automatic incrementation of a score etc.
- **Is the timer equal to a certain value?** This condition is true when the timer reaches the given time. The timer is initialised to zero at the start of the frame.



## Conditions of the create object

- **Pick a random object.** This powerful condition will select one object in the middle of all the objects.
  - **Pick an object at random in a zone.** This condition is like the previous one, but limited to the object inside a given zone on the frame.
  - **Pick all objects in a zone.** Will select all the objects present in a zone on the frame.
  - **Pick all objects on a line.** Will ask you to enter the co-ordinates of two points. This condition will draw a hidden line between the two points, and select the objects that cross this line.
- Note: for the Pick object conditions, the actions will only apply to the objects selected by the condition.



## Conditions of the player object

- **Read joystick state.** This condition allow you to test the state of the joystick: know if the user has pressed up, down left or right, along with the fire buttons.
- **Repeat while joystick is pressed.** This condition will be true as long as the player presses the wanted key on the joystick.



## Conditions of the active object

The active object is very versatile, and it contains a lot of conditions. Here are the main ones...

- **Collisions with another object.** Important condition, it is true immediately after the object has been colliding with another. Use it in games, to make a ball bounce etc.
- **Collisions / Overlapping another object.** This condition is constantly true as long as the objects are overlapping each other.
- **Collision / Backdrop.** This condition will be true if the object is colliding with a background object (Backdrop or Quick backdrop) marked "Obstacle" or "Platform".
- **Collision / Overlapping a backdrop.** This condition is constantly true if the object is overlapping a background object marked as "Obstacle" or "Platform"...

- **Position / test position of object.** This condition will ask you select which border of the frame to test, and it will be true if the object goes through the given border.
- **Alterable values / Compare to one of the alterable values.** Allow you to compare a value to one of the alterable values. If the comparison is true, the condition is true.
- **Alterable values / Compare to one of the alterable strings.** Allow you to compare a string to the content of one of the alterable strings.



## Conditions of the counter object

- **Compare the counter to a value.** Allows you to make a comparison with the value of the counter. If the comparison is true then the condition is true.

## Actions of interest

The number of actions in Multimedia Fusion is even greater than the number of conditions. There are actions for every object, in every domain. Refer to the electronic documentation for an exhaustive list of all the actions. We will concentrate on the main actions for the principal objects.



## Special object actions

- **Change a global value.** Allows you to modify the value of a global value.
- **Set global string.** Changes the content of one of the global strings.
- **Group of events / Activate.** Activates a group of events: all the events within the group are tested again. The **On group activation** condition is also called.
- **Group of events / Deactivate.** Deactivate a group of events: all the events within the group will become inactive.
- **Randomise.** Initialise the value of the random generator. Random numbers are generated from a mathematical calculation. If you set the randomise seed to a specific value, the list of numbers generated will always be the same: very useful to make demos of your product where you expect the numbers to be the same at each session.
- **Fast loops / Start loop.** Initiate a fast loop. More info in appendix 3.



## Speaker object actions

The speaker object actions allow you to play music and samples.

- **Play sample.** Opens the sound selector for you to choose a sample. This sample can be part of another application loaded in Multimedia Fusion, an application on disc, or a simple WAV file on disc. If you check the “Uninterruptible” checkmark, the sample will not be interrupted until it ends.

- **Play and loop sample.** Same as above, but also asks for a number of times to play the sample.
- **Play music.** Allows you to select a MIDI music to play using the music selector.
- **Play and loop music.** Same as above, but asks for a number of time to play the music.



## Storyboard object actions

The actions of the storyboard object control the flow of the application.

- **Next frame.** Quits the current frame and loads (and plays) the next one in the storyboard list.
- **Previous frame.** Quits the current frame and loads (and plays) the previous frame in the storyboard list.
- **Jump to frame.** Asks for the frame to jump to. You can also do a calculation to get the number of the frame.
- **Restart the current frame.** As the name indicates, restarts the current frame: all the objects, movements, values are reset to their original values.
- **Restart the application.** Restarts the application from frame 1 as if it were run for the first time.
- **End the application.** Immediately quits the application.



## Mouse pointer and keyboard object actions

- **Hide Windows mouse pointer.** Hides the pointer.
- **Show Windows mouse pointer.** Shows the pointer if it was hidden.



## Timer object actions

- **Set timer.** Changes the timer of the frame to a new value.

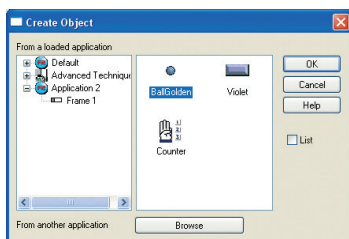


## Create object actions

This object only contains one action, but it is an important one.

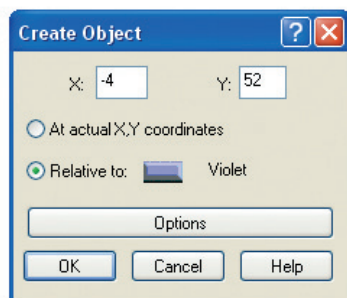
### Create object

This action allows you to create new object during runtime. When you choose this option, you are first presented with an object selector in which you have to choose the object to create.



As a default, the box shows the objects defined in the **current** frame. The left pane of the box displays the applications loaded in memory at the time. You can open the application and open each frame and see the list of objects of each frame. You can also click on the “Browse” button to load the object list contained in an application on disc. Once the object is chosen, double click on it to proceed further.

The frame is now displayed, with a small dialog box and a flashing cross in the middle of the frame.

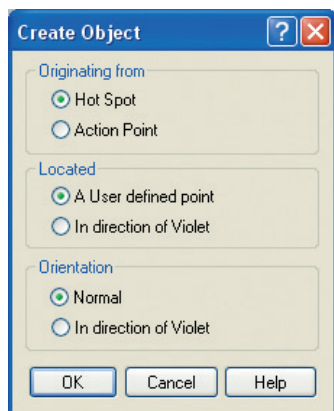


- The **cross** represents the **position of creation** of the new object. The X and Y co-ordinates are displayed in the box. You can move the cross on the frame with the mouse pointer. This box contains **two modes** of functioning: **absolute position** and **relative position**. Absolute position: the object will be created at the given co-ordinates. Relative position: the object will be created at a certain distance from another object.
- To go into relative positioning, simply click on the object in the frame, or click on “**Relative to**” and select the object to use. A relative position is represented by a link between the cross and the selected object. Once in relative mode, you have access to an option screen:

• **Originating from.** Indicates where the origin point of the relative object is taken. It can be from the hot-spot, or from the action point.

• **Located at a user defined point** means that the point is calculated by adding the offsets to the position of the object. **Located in direction of “object”** enforces Multimedia Fusion to take the rotation of the relative object into account and rotate the coordinate accordingly. Imagine a tank seen from above, you can turn it in every direction, with this option the bullets will follow the movement of the gun.

• **Orientation Normal:** will use the default orientation of the created object. **In direction of “Object”** will equal the direction of the created object to the direction of the relative object: in our tank example, the bullet would be created in the direction of the gun.





## Player object actions

- **Score / Set score.** Allows you to change the score of the player. You can also add to, or subtract from the score. The score will be reflected in a score object if it exists on the frame.
- **Lives / Set number of lives.** Allows you to change the number of lives of the player. The number of lives will be reflected in an eventual lives object on the frame.

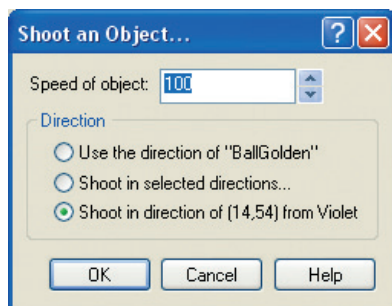


## Active object actions

The active object contains a lot of actions, designed to control the movement and animations. Most of the actions are self-explanatory, let's concentrate on some important actions.

- **Movement / Stop.** Stops the movement.
- **Movement / Start.** Restarts a stopped movement.
- **Movement / Bounce.** Makes the object bounce when colliding with another object or an obstacle.
- **Direction / Set direction.** Allows you to change the current direction of the object. You can if you want calculate the new direction or set it with the direction selector.
- **Direction / Look in direction of...** Makes the object face another object. If you do it constantly, you can create intelligent bullets that look for their target.
- **Shoot an object.** This action is rather frequently used in games. It allows you to shoot another object. First of all you have to indicate which object to shoot, using the same dialog box as for the Create Object action. Once this done another dialog box appears:

- **Speed of object.** Enter the speed of the bullet.
- **Use the direction of "object",** if selected the bullet will be shot based on the direction of the source object, just like our little tank above.
- **Shoot in selected directions.** Opens a direction selector in which you can define the directions to use. If more than one direction is selected, then a random direction within the selected ones is chosen.
- **Shoot in direction of...** opens the position selector as for the Create object action, for you to choose a point in the frame (it can be relative to another object).



**Important note:** as stated in the animation editor chapter, the bullets are shot from the action point of the parent object: be sure to define these action points correctly for a better visual effect. Also note that the **shooting animation** of the parent object is started when the action is executed. If it is defined, you obtain a nice effect on screen.

- **Alterable values / Set.** Changes one of the alterable values of the object.
- **Alterable strings / Set.** Changes one of the alterable strings of the object.
- **Destroy.** This very important action allows you to destroy the objects instantly. If a “Disappearing” animation is defined, it will be automatically played before the destruction of the object. If a fade out transition is defined also, it will be played before destruction.



## Counter object actions

Counter are moveable objects: you will find in them all the movement control actions that you can find in the active object. Let's concentrate on the actions specific to this object:

- **Set counter.** Changes the value of the counter. If the counter is visible, the value is reflected in the display.
- **Set minimum value.** Changes the minimum value of the counter.
- **Set maximum value.** Changes the maximum value of the counter.



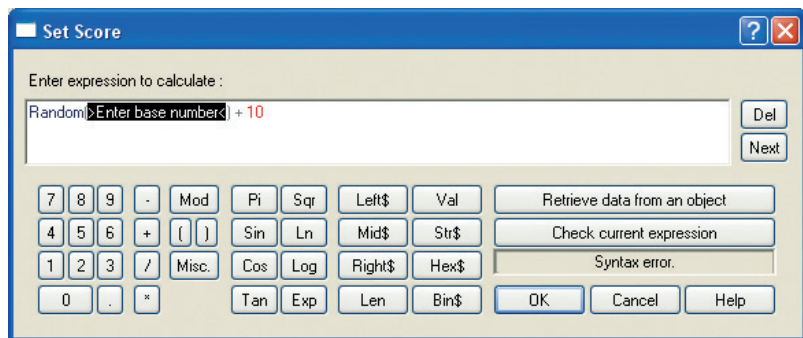
## String object actions

The string object is also a moveable object, with all the according actions. Here are the most important actions more specific to this object:

- **Change alterable string.** Modifies the string contained in the object. The result is immediately displayed on the screen if the object is visible.
- **Set paragraph.** Changes the current paragraph displayed by the object.

## 8. The Expression Editor

Many actions and conditions in Multimedia Fusion ask for you to enter a number or a calculation. In order to do that, they present you with the Expression editor. The expression editor is a powerful tool that enables you to make calculations, even complex ones, and retrieve information from objects, like their coordinates and include them in the calculation



The expression editor looks like a **calculator**. It consists of:

- An **edit zone** in which you can type your calculation
- **Number** buttons to enter the digits
- **Operator** buttons like +, -, \*, /, Mod and brackets to perform your calculations
- **Mathematical functions** like sine and cosine
- **Text functions** like Left\$, Right\$ etc.
- A **Retrieve data from object** button to gather information from existing objects in the application.
- A **Check current expression** button that tests the validity of the expression. It displays "Valid expression" if the expression is correct and "Syntax error" if not.

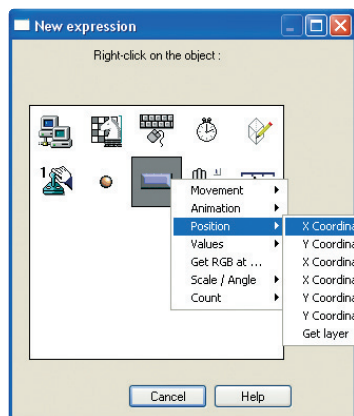
The expression editor allows you to make calculation with numbers, but also with strings. A string is a chunk of text included in **quotes**. For example "My string" is a string and will be accepted by the editor. You can perfectly add two strings together, for example: "This is" + " my string" will return in one string equal to "This is my string". The other operators, like multiplication and division are not accepted for strings.

Some actions ask for a number, in this case the expression editor will be initialised with a zero, and some ask for strings, in this case it will be initialised with an empty string "". If you mix numbers and strings together, the expression editor will report a "Mixed strings and number" error, and will refuse to be closed.

A closer look on certain functions:

- **Pi**. Returns the value of the Pi number.
- **Sin / Cos / Tan**. Return the value of the sine, cosine and tangent of an angle. The angle must be expressed in degrees (0-360).
- **Sqr**. Returns the Square root of a positive number.
- **Ln, Log**. Return the logarithm of a number.
- **Exp**. Returns the exponential of a number.
- **Left\$**. String function, returns the left part of a string. For example Left\$( "Multimedia", 3 ) will return "Mul".
- **Mid\$**. String function, returns a extract from the middle of a string. This function asks for two parameters, the first is the start position in the string, the second is the number of characters to return. Example, Mid\$( "Multimedia", 4, 4 ) will return "time".
- **Right\$**. String function, returns the right of a string, for example Right\$( "Multimedia", 3 ) will return "dia".
- **Len**. Returns the length of string, example Len( "Multimedia" ) will return 10.
- **Val**. Converts a string into a number, example Val( "1000" ) will return the number 1000.
- **Str\$**. The contrary of the previous function, converts a number into a string. Example, Str\$( 1000 ) returns the string "1000".
- **Hex\$**. Converts a number into a string containing the hexadecimal value of the number. Example Hex\$( 256 ) returns "100".
- **Bin\$**. Converts a number into a string containing the binary value of the number. Example Bin\$( 16 ) returns "10000".

## Retrieving data from objects



This very important function of the expression editor opens multiple possibilities of programming. A click on the button "Retrieve data from objects" opens a dialog box that presents you with the objects of the application.

You should click with the right mouse button on one the objects to open the function pop-up menu of the object...

Once the menu option is chosen, the box closes itself and the token of the function is inserted in the edit zone of the expression editor. You might have to enter one or more parameters for

the function: the parameter is automatically selected.

A click on the "**Next**" button will select the next parameter for entry.

## Some of the available functions

As for actions and conditions, we are not going to review all the available functions of all the objects: refer to the electronic documentation for an exhaustive list.



### Special object functions

- **Generate a random number**, =Random( X ). This function, very important for games, will return a random number ranging from 0 and to X-1. X must be lower than 32768.
- **Retrieve a global value**. Returns the value of a global value.
- **Retrieve a global string**. Returns the content of a global string.



### Storyboard object functions

- **Current number of frame**. Returns the number of the current frame.
- **Frame / Width**. Returns the width in pixels of the frame.
- **Frame / Height**. Returns the height of the frame in pixels.



### Mouse pointer and keyboard object

- **Current X Position of the mouse**. Returns the horizontal position of the mouse in the frame. This co-ordinate can be negative or larger than the width of the frame if the mouse pointer is outside of the frame.
- **Current Y Position of the mouse**. Same as above, but vertically.



### Timer object functions

- **Timer count, in 1/1000 since start of frame**. As its name indicates...
- **Timer count, hours / minutes / seconds / 1/100**. Returns the value of the timer since the start of the frame.



### Create object functions

- **Total number of objects**. Returns the count of all the objects in the frame. Useful for debugging your application.



### Player object functions

- **Current value of score**. Returns the score of the player.
- **Current number of lives**. Returns the number of lives of the player.



## Active object functions

- **Movement / Speed.** Returns the speed of the object from 0 to 100.
- **Animation / Current direction value.** Returns the number of the direction of the object, from 0 (to the right) to 31.
- **Position / X Co-ordinate.** Returns the X coordinate of the object (the position on screen of the horizontal hot spot of the image).
- **Position / Y Co-ordinate.** Same as above, vertically.
- **Values / Alterable values.** Returns the content of one of the alterable values.
- **Values / Alterable strings.** Returns the content of one of the alterable strings of the object.
- **Count / Number of object.** Returns the number of this kind of object created on the frame.



## Counter object functions

The counter object is a moveable object: the function menu contains all the default movement functions. Let's concentrate of the functions specific to this object:

- **Current value.** Returns the value of the counter.
- **Minimum value.** Returns the minimum value of the counter.
- **Maximum value.** Returns the maximum value of the counter.



## String object functions

The string object is also a moveable object, it contains movement functions. The specific functions are the following:

- **Alterable string.** Returns the string contained in the object.
- **Text of a paragraph.** Returns the text of one of the paragraphs defined in the string object. This asks for one parameter, the number of the paragraph to be returned.

## 9. The Event-list Editor

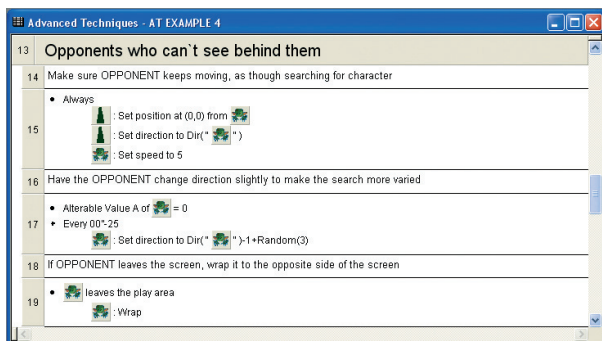
The event-list editor is the fifth main editor contained in Multimedia Fusion. As with the event editor, it allows you edit the events of the application. The difference is in the presentation of the events: they are presented as a **list**, not as a grid. Some people prefer to use the event-list editor.

**Note:** the Event List editor is not included in The Games Factory, it is only present in Multimedia Fusion.

### How to reach the event-list editor

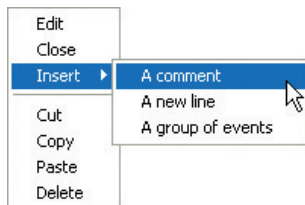
There are multiple ways of opening the event editor.

- In the **workspace toolbar**, right click on a frame, and choose event-list editor
- In the **storyboard editor**, right click on a frame and choose event-list editor
- Click on the **event-list editor button** in the navigation toolbar
- Press CONTROL-L



As you can see on the picture, the events, the remarks and the groups are listed one after the other. A click on the number of the event opens the following pop-up menu:

- **Edit.** Edits the current line, a group or a remark.
- **Close.** Only available for groups: closes the current group. If you click on a closed group, this option is replaced with "Open".
- **Insert / A comment.** Inserts a new remark in the code.
- **Insert / A new line.** Opens the New condition dialog for you to choose a new condition. A new line of events is then created.
- **Insert / A group of events.** Opens the New group dialog for you to enter the name of a new group. This one is created as soon as you close the box.
- **Cut / Copy / Paste / Delete.** Usual clipboard functions.



A click with the right mouse button on a condition or an action opens this menu:

- **Edit.** Edits the parameters of the action or the condition.
- **Replace.** Replace the action or condition by a new one.
- **Insert.** Inserts a new action or condition.
- **Negate.** Negates the condition, if possible.
- **Cut / Copy / Paste / Delete.** The usual clipboard functions are available in the event-list editor.

|         |  |
|---------|--|
| Edit    |  |
| Replace |  |
| Insert  |  |
| Negate  |  |
| <hr/>   |  |
| Cut     |  |
| Copy    |  |
| Paste   |  |
| Delete  |  |

**Drag & drop** is, like in the event editor, an important part of the event-list editor. You can drag & drop conditions, actions, remarks, groups etc.

## Printing the events

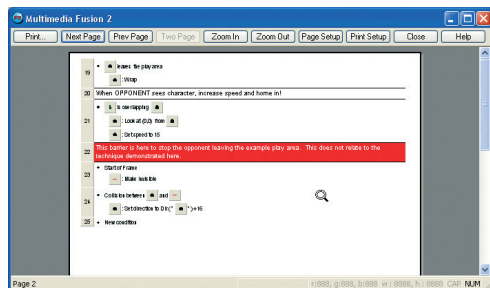
The main advantage of the event-list editor is the ability to print the events. You will find the print options in the **File** submenu of the main menu.

- **Print.** Prints the current list of events.
- **Print preview.** Opens the print preview page for you to verify the format of printing.
- **Page set-up.** Opens the page set-up dialog box.

## Print Preview

The print preview page shows you the events as they are going to be printed. The different buttons on the top of the window are the following:

- **Print.** Launch the printing.
- **Next page.** Displays the next page of events.



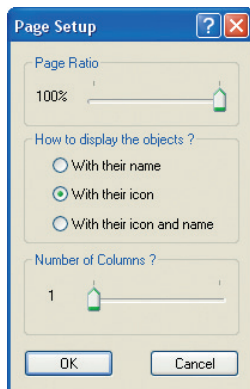
- **Previous page.** Displays the previous page of events.
- **Two pages.** Displays two pages instead of one.
- **Zoom in.** Magnify the display.
- **Zoom out.** Diminish the zoom factor.
- **Page Setup.** Opens the page set-up dialog.
- **Print Setup.** Opens the printer set-up page.

- **Close.** Returns to the event-list editor.

## The Page set-up dialog

This dialog allows you fix a certain number of parameters for the printing.

- **Page ratio.** Sets the number of events printed on a page.
- **How to display the objects?** With their name will print only the name of the objects. With their icon will print the icon (maybe difficult to read once printed). Final and best option: with their icon and name.
- **Number of columns.** Allows you to select the number of columns of events in one page.



## Appendix 1. Global events and behaviors

In this manual we have detailed the events associated with a frame. They are the main events. But you can if you want define other events that might be useful to your application.

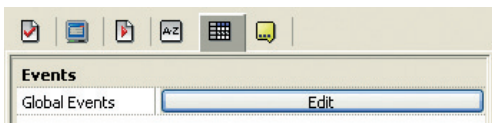
### Global Events

If you create an application with multiple frames with the same interface (like a presentation with a previous / next button), you might not want to repeat the same list of events in every frame: not only inserting them would be tedious, but if you decide to modify the way of working you would have to port the changes in all the frames.

This is why Multimedia Fusion contains the possibility of creating **global events**.

Global events are a list of events that is **global** to the application: it will be executed in **all the frames** of the application. In our presentation example, you would put in the global event the handling of the next / previous buttons.

To open the global events, display the properties of the application (by clicking on the name of the application in the workspace toolbar) and click on the “**Events**” tab.



A click on the “**Edit**” button will open the event editor for the global events. This editor is the event editor we have seen before but there is a slight difference: the objects. You will notice upon opening the editor that only the system objects are presents, no object of the application is presented to you. You have to Import the objects in the global event editor to use them.

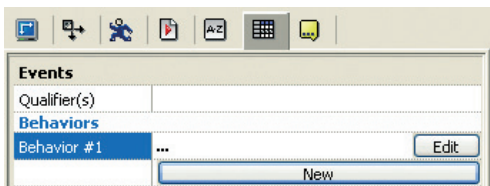


Look at the object bar, you will notice one new object though, named “**Import new objects**”. A click on this object will open a dialog box presenting you with all the objects of all the frames of the application (this might make a lot of objects). Double click on the object you want to import: the dialog is closed and the object is present in the editor. You can now use it as a normal object. You will notice a small arrow in the right corner of the icon of the object, indicating that the object has been imported.

## Behaviors

Behaviors could also be called “Local events”. Behaviors are list of events contained in one object. The events are automatically activated in the frame if the object is present. Behaviors allow you to create “Intelligent” objects that will react as soon as they are dropped in the frame, without the need to define events in the frame list of events.

To create a behavior, you must first display the properties of the object in the property toolbar: just click on the object in the frame editor. Then click on the **“Events”** tab.



Click on the **“New”** button to create a new behavior. You can have more than one behavior per object. You can rename it by clicking on it. Click on the **“Edit”** button to open the event editor for this behavior.

As with the global event editor, the behavior event editor is the classic event editor you have seen before, but not all the objects are present in the object bar. You will find the object containing the behavior and the “Import new object” object. As with the global events editor, a click on this object presents to you all the objects of the frame for you to import them.

Note: when you are editing the events of the frame, the objects containing a behavior are marked by a little “B” in the corner right of the object’s icon. If you click with the right mouse button on this object in the object bar, you will have an option to open the behavior of the object and edit it.

Global events and behaviors can become an important part of your application. They can help to reduce dramatically the number of frame events needed to handle each frame.

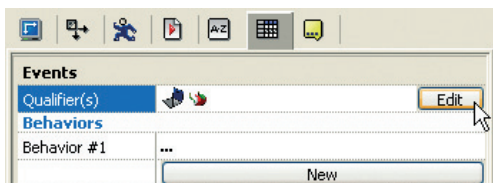
## Appendix 2. The qualifiers

Qualifiers are a way of reducing the complexity of the list of events of the frame. Imagine you create a game in which the player must avoid a lot of different traps. Without qualifiers, you would normally have to detect the collision between the player and trap for each trap: if you have a lot of those, this could create a lot of event lines.

A qualifier is a symbol that you associate with an object. In our example, you would associate the qualifier **“Traps”** to every one of the traps in your game. Once in the event editor, you would see a new icon, the Traps qualifier. You could then replace all the collision lines by one line, the detection of the collision between the player and the qualifier called Traps.

Qualifier can be used exactly as the object they replace: they can have conditions, actions and functions. Actually there is no difference between an object and the qualifier that represents it.

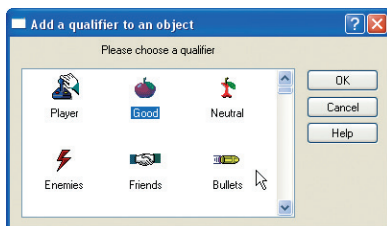
To add a qualifier to an object, you first need to display its properties in the property toolbar. Just click on the object in the frame editor. Then choose the **“Events”** tab (the same than the one with the behaviors).



Click on the **edit** button to open the Qualifier dialog box.



This box presents the qualifiers already selected for the object. A click on the **“Add”** button allows you to choose a qualifier amongst the available list of qualifiers:



**Conclusion:** qualifiers can save you a lot of time developing the application, reduce the amount of code and make your list of events easier to read.

## Appendix 3. Fast loops

Many times in your program of events, you need to repeat the same action a certain number of times. There are a number of ways of achieving this: you can use the “Repeat” condition from the special object menu, or use a counter to count the number of repeats.

Multimedia Fusion provides you with a safe and very fast way of repeating the same actions: the **fast loops**.

### What is a fast loop?

A fast loop is a system that will repeat the same events a given number of times, at fast speed.

### How to create a fast loop?

First of all you need to initialise the loop. Open the action pop-up menu of the special object, and open the **Fast loop** submenu. You will find an action called “**Start loop**”. This action asks for two parameters:

- **The name of the loop.** This must be a string depicting the name of your loop, choose it short and easy to recognise.
- **The number of times to loop.** Enter the number you wish here.

Now you need to create a new “**On loop**” condition. Click on **New condition**, and open the condition pop-up menu of the special object. In the bottom of the menu you will find an “**On loop**” entry. Choose it. The program now asks for the name of the loop, enter the same name as before. You now have a new line of condition.

Now enter the actions you want to execute after this condition.

### How does it work?

When Multimedia Fusion encounters the **Start loop** action, it searches for the **On loop** conditions of the program. If it finds one, it compares the name of the loop, if it is the same, it executes the actions of the event (if not it continues the search). Once the actions are executed, it increments the loops counter and repeats the actions: the actions are called at high speed the exact number of times you entered.

- You can have more than one “On loop” conditions in the program for the same loop.
- You can retrieve the number of the loop in an expression: a “**Get loop count**” function exists in the special object functions menu.
- You can also create a loop within a loop creating a nested loop. But be careful about the processing time of such nested loops as they it can be very long.

## Appendix 4. Creating a scrolling game

Scrolling games are fun to play and it is very easy to create a scrolling game in Multimedia Fusion. Here are the different steps to follow.

### Simple scrolling game

- Change the size of your application window (in the application properties) to a reasonable size, like 640x480 or 320x200 for fast scrolling.
- Set the size of your frame **larger** than the window of the application. For example, if your application window is 320x200 you can set the size of the frame to 960x200 and you will enough space to create a horizontal scrolling three times the size of the window.
- The scrolling starts at X position zero (the far left edge of the frame): position your characters, scores, number of lives in the top left of your frame.
- In the event editor, create an **“Always”** condition, and enter the next action: open the storyboard object's pop-up action menu and choose the action **“Scrolling / Center Window position in frame”**, and choose to center the position on your main character.

Voila! If you run the frame, you will see that the screen automatically scrolls and follows the character.

### Static objects

As a default, active objects will follow the scrolling: their position will remain static relative to the scrolling background. But for certain objects, like the score and lives objects, you need them to stay at the same position on the screen and ignore the scrolling. This can be done by un-checking an important property of the objects. To access this property, click in the frame editor on the object, and in the property toolbar click on the **“Runtime options”** tab. Uncheck the property name **“Follow the frame”**: your object will ignore the scrolling from now on.

### Parallax scrolling game

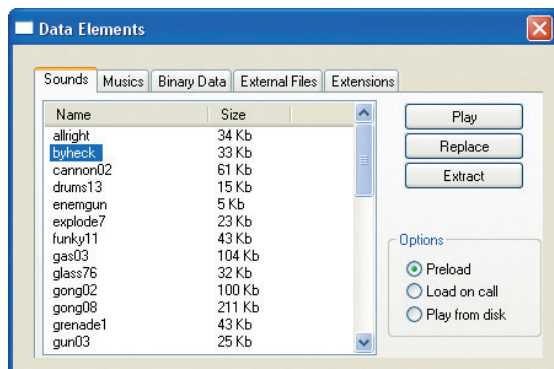
Parallaxing is best described as distant object moving slower than close objects. Imagine you are driving down the road and looking out a side window. Objects in the far distance (mountains, the moon, stars) appear to move very slowly or not at all. Objects at a mid distance (hills, buildings, trees) move by more slowly than close objects such as the road signs. This perceived appearance of movement is called Parallaxing. You can create parallax scrolling with Multimedia Fusion quite simply.

- Set the application window size and frame size as explained before.
- In the frame editor, open the **Layer toolbar**, and create as many layers as you need of scrolling planes.
- Drop the objects in the planes to create your display.
- Click on each layer in the layer toolbar to open its properties in the property toolbar. You will notice two properties: **X Coefficient** and **Y Coefficient**. Each of them represents the speed of scrolling of this layer relative to the position in the frame. If the number is below 1, the layer will be slower. For example, a coefficient of 0.5 will make the layer scroll twice as slow as the main layer. A coefficient of 2 will make it scroll twice as fast.
- You should always have you main character and action in the layer with a coefficient of 1
- Eventually check the **“Wrap”** property for layers that go faster than the main layer to ensure the graphics are always present on the screen.
- Now go in the event editor and enter the **“Center window position in frame”** action.

## Appendix 5.

# Editing the data elements of an application

If you click with the right mouse button on the name of an application in the workspace toolbar, you will find an option called “**Data elements**”. This very powerful option allows you to manage the data contained in your application. Choose this option and it opens the Data element dialog box.



The purpose of this box is to show and edit all the elements of data included in the application. It is a little like “opening the hatch” and seeing the inners of your application. These data can be sounds, music, and binary data.

The previous picture presents the sounds panel. All the samples used in the application are listed in the box. You can listen to them with the **Play** button. The **Replace** button allows you to choose another sound and replace an existing sound by the new one. This is a very powerful option as all the frames using this sound will use the new sound, you do not need to edit the events to replace the sample.

Extract allows you to save the sound on disc to re-use it later: if you are given a game with cool sounds, you can extract the sounds and use them in our own game (copyright allowing of course).

Some options are available for each sound:

- **Preload**. This is the default option, the sound is loaded in memory when the frame starts. Pros: fast to play. Cons: if you have a lot of samples in your frame, loading the frame may take a little time.
- **Load on call**. The sound is only loaded when it needs to be played. Pros: faster loading time for the frame. Cons: may induce a little delay just before playing the sound.
- **Play from disk**. The sounds are not loaded, they are played directly from the disk.

Pros: you must use this option for large samples (several Megabytes). Cons: here too many induce a small delay before playing the sound.

The Binary tab allows you to embed files into your application. These files can be opened directly from the application by objects that need them. This allows you for example to protect graphic files played by the Active Picture object, or a text file displayed by a RichEdit object. Note: some objects like the video objects will extract the file to a temporary file on the hard disc before playing it, and not all the 3rd party extensions use this feature. The Binary tab is only present in Multimedia Fusion.

The extension tab presents to you all the extension objects used by the application.

# Appendix 6. Extensions

You have seen in this documentation that Multimedia Fusion contains a lot of different objects. Not all of these objects are included in the main program, most of them are stored in **external files**, and are called extension objects.

The internal objects are:

- The Backdrop object
- The Quick Backdrop object
- The Active object
- The Counter object
- The String object
- The Score object
- The Lives object
- The Formatted text object
- The Question and Answers object
- The Sub-Application object

All the other objects are external objects, their filename ends with the suffix **“.MFX”**. They are stored in two folders on your disk:

- The **edit time** version of the objects are stored in the **“Extensions”** folder located near the main Multimedia Fusion executable.
- The **runtime** version of the objects is stored in the **“Data \ Runtime”** folder. You will notice that the size of the objects in this folder is smaller than in the Extensions folder: they only contain the runtime code. It is important not to mix the objects of the two folders.

## New transitions

The fade transitions that you use on frames, are also stored as external files in the **“Transitions”** folder beside Multimedia Fusion main executable.

## New movements

The system of extensions exists also for movements. For example, the Pinball movement is an external file. The new movements are files with a **“.MVX”** suffix, and are stored in the **“Extensions”** folder. Extension movements are only available in Multimedia Fusion.

Developers are creating new objects, new movements and new transitions regularly. You can download them from the internet: come often to [www.clickteam.com](http://www.clickteam.com) to find if new objects, movements or transitions are available. Clickteam also publish every so

often “Bonus packs” that contain many new files.

If you are fluent in C++ development you can create your own objects: come to the Clickteam web site and download the MMFSDK, the Multimedia Fusion development kit available in the download section of our site.

Extension objects can only be created for Multimedia Fusion. The Games Factory extra objects will be published by Clickteam.

# Appendix 7. The runtime options

If you display the properties of an application (by clicking on the application in the workspace toolbar), you will notice a “**Runtime options**” tab. This page contains important properties to fine tune the runtime.

**Runtime options**

Frame Rate **50**

☐ Machine-independent speed

☐ Run when minimized

☐ Run while resizing

☐ Do not stop screen saver when input event

**Sound options...**

☐ Multi-samples

☐ Play sounds over frames

☐ Do not stop samples when application loses focus

**Display options...**

Display Mode **Standard**

☐ V-Sync

☐ Enable Visual Themes

**Players...**

Initial score **0**

Initial # of lives **3**

Default Controls **Edit**

**Vitalize**

Preview

- **Frame rate.** Set the number of frames per second of the runtime. The default value is 50, it can go from 1 (one step of animation per second) to 1000 (twenty times as fast as the default settings).

- **Machine independent speed.** This option is specially designed for heavy graphics games, with big objects that take some time to be drawn. If you notice a slow down in the animation (meaning the runtime cannot run at 50 frames per seconds), it might be worth to try this option.

- **Run when minimised.** If unchecked, the runtime will go in pause mode when the window is minimised. Check this box to make it continue to run.

- **Run while resizing.** As a default, the

runtime is in pause while you are resizing the window. Check this box to make it continue to run.

- **Do not stop screen saver when input event.** You might want to create interactive screen savers, like small games. Use this option and the screen saver will not vanish when the user presses the keyboard.

- **Sound / Multi samples.** Allows more than one sample to be played at the same time.

- **Sound / Play sound over frame.** If checked, the sounds (samples and music) will continue from one frame to another.

- **Sound / Do not stop samples when application looses focus.** The default attitude is to cut the sound when you activate another application.

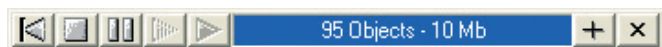
- **Display / Display mode.** This option allows you to choose the method of displaying the images used by the runtime. **Standard** uses normal display routines and work on every machine. **DirectX** uses Microsoft DirectX interface. It is usually slower than Standard but you might need it if your application uses the Change Resolution Mode option and has a non-standard resolution (e.g. 320x200 is not available on all the machines with the **Standard** display mode). **DirectX + VRAM** is usually faster, especially for scrolling applications, but it can be also very slow with some ink effects or transitions, so it depends on your application.

- **V-Sync.** Will synchronise the display with the vertical synchro of the monitor, resulting in a clearer display.
- **Enable visual themes.** With this option, your application will use the Visual Theme defined by the user in its Windows Screen properties. Without this option, your application will be displayed using the classic Windows style.
- **Players / Initial score.** If you want the scores of the player to be different from zero when the application starts.
- **Players / Initial # of lives.** Enter the number of lives at the beginning of your game.
- **Players / Default controls.** Click on the Edit button to modify the default controls of the player. The initial setting is all four players on the keyboard.
- **Vitalize / Preview.** Choose a picture to display while the Vitalize! Plugin is loading your application.

## Appendix 8. The debugger

When your application gets complex, with a lot of objects and a lot of events, sometimes it does not work as you want, due to errors in the logic applied. Multimedia Fusion contains a tool to help you find the problems of your application: the **debugger**.

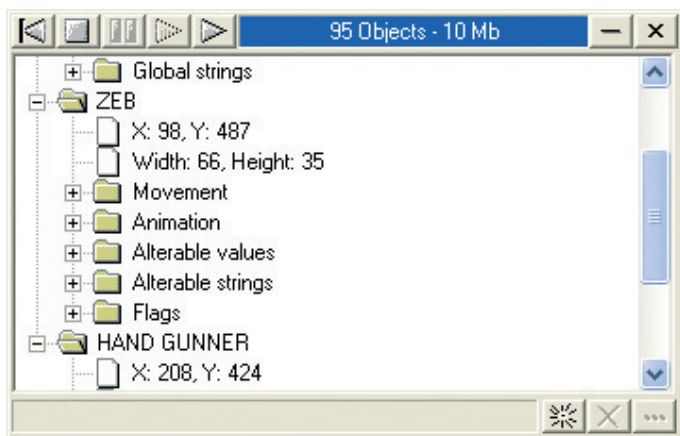
The debugger is automatically opened when you launch the application. It is a small bar in the top left corner of your screen.



This little bar does not interact with your application. It takes no processing time and just sits there until you touch it. From the left to the right:

- **Rewind.** Restarts the current frame.
- **Stop.** Quits the application and comes back to Multimedia Fusion editor.
- **Pause.** Goes into pause mode: in this mode you can add objects and see their values. See later.
- **Step by step.** Perform only one step of animation allowing you to see what exactly is happening in the application.
- **Play.** Resume to normal speed.
- **Information zone.** This zone indicates permanently the number of objects contained in the application, and the memory size necessary to make it run.
- **+ button.** Opens the debugger window.
- **Close.** Closes the debugger, but keep the application running.

If you open the debugger by clicking on the + button, you obtain the following window:



As you can see this window displays **information**. Information about what is going on within your application. The display is centred on the notion of **objects**. The initial object displayed is the system object that contains information about the frame rate, the number of the frame, and the global values and strings. All the information is displayed as a tree, you can open the different sections with the mouse.

\* You can add objects of the application to the display. First you need to go in **Pause** mode by clicking on the **Pause** button. Then you have two options:

- Click on the “**Add object**” button in the bottom right of the window and choose the object in the list

- **Directly** click on the **object in the frame**, it will be added automatically to the debugger.

\* You can keep the debugger open and **restart the application**: the value will be updated as the application changes.

\* You can also change the value of alterable values or strings by **double clicking** on them.

With the debugger you can monitor an object while the application is running. The monitored object does not have to be displayed on screen or interacted with at all.

## Appendix 9.

# Creating your standalone application

The goal of Multimedia Fusion is to create applications that you can distribute around you, give to friends and publish on the internet. This kind of application needs to work alone on a computer, without Multimedia Fusion installed: this is the reason why they are called standalone applications.

Multimedia Fusion permits to create three kinds of standalone applications:

- **Executable files** (.EXE) that run when you double click on them
  - **Screen savers** (.SCR) that automatically pop-up when you leave your computer unattended
  - **Internet applications** (.CCN), that you can play on a web page with the **Vitalize!** Plugin.
- Note that CCN files are the ones understood by the sub-application object, so if you have a project in which sub applications are used, you should compile in this format.

To choose what kind of application you want to create, display the application properties (by clicking on the application in the workspace toolbar), and click on the “Settings” tab. You will notice the following properties:

|                         |                                     |
|-------------------------|-------------------------------------|
| <b>Build</b>            |                                     |
| <b>Build Type</b>       | <b>Stand-alone Application</b>      |
| <b>Build Filename</b>   |                                     |
| <b>Install Settings</b> | <input type="button" value="Edit"/> |

- **Build type.** Choose between the three possibilities (application, screen saver or internet).
- **Build filename.** As a default (if this property is left empty), the name of the built file will be the name of your application with the appropriate suffix (.EXE, .SCR or .CCN). You can if you want change this with this property.
- **Install settings.** If you decide to create an installer, click on this button to edit its settings.

To build your application, use the menu option **Build / Application** in the **File** menu. To build an entire project, choose the **Build / All** option.

## ***Appendix 10. Vitalize!***

Vitalize! Is an Internet plugin that plays Multimedia Fusion applications, built as Internet applications, directly in a Web browser (MS Internet Explorer, FireFox, etc.). Please refer to the electronic on for more information.

# Notes